

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

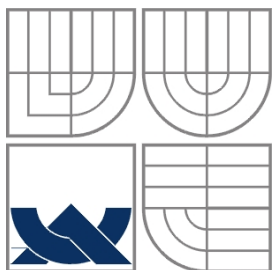
INFORMAČNÍ MODULY KOMUNIKUJÍCÍ PŘES BLUETOOTH

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

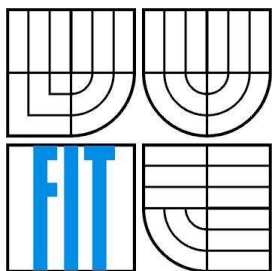
AUTOR PRÁCE
AUTHOR

MAREK KNYTL

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

INFORMAČNÍ MODULY KOMUNIKUJÍCÍ PŘES BLUETOOTH

INFORMATION MODULES SUPPORTING BLUETOOTH COMMUNICATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MAREK KNYTL

VEDOUCÍ PRÁCE

SUPERVISOR

ING. LUKÁŠ MIČULKA

BRNO 2011

Abstrakt

Tato práce popisuje vytvoření dvou Informačních modulů, které jsou řízeny mikrokontrolérem a ovládány pomocí klávesnice a displeje. Tyto zařízení komunikují se serverem, běžícím na PC, pomocí technologie Bluetooth a jsou určeny pro vzájemné zasílání krátkých textových zpráv. Úvod je věnován teorii ohledně Bluetooth, dále přechází k jádru práce, což je návrh a realizaci hardwarové části Informačního modulu a vývoj firmware pro jeho řízení. Nakonec se věnuje vývoji serverové aplikace pro PC.

Abstract

This bachelor thesis describes the creation of two information modules, which are controlled by microcontroller and command using keyboard and display. These devices communicate with server, running on PC, using Bluetooth technology and they are designed for mutual transmission of short text messages. The introduction is devoted to the Bluetooth theory, than goes to the core of this bachelor thesis, which is the design and implementation of information module hardware and its firmware development. Developing server application for the PC is mentioned at the end of this document.

Klíčová slova

Bluetooth, RFCOMM, Bezdrátová komunikace, AVR

Keywords

Bluetooth, RFCOMM, Wireless communication, AVR

Citace

Knytl Marek: *Informační moduly komunikující přes Bluetooth*, bakalářská práce, Brno, FIT VUT v Brně, 2011

Informační moduly komunikující přes Bluetooth

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Lukáše Mičulky

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Marek Knytl

9. května 2011

Poděkování

Děkuji vedoucímu mé bakalářské práce panu Ing. Lukáši Mičulkovi za odbornou pomoc a ochotu při zařízení objednávky součástek. Dále bych chtěl poděkovat panu Ing. Václavu Šimkovi za výrobu desek plošných spojů pro informační moduly.

© Marek Knytl, 2011

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod.....	3
2	Bluetooth.....	4
2.1	Verze Bluetooth, podle historického vývoje.....	4
2.1.1	Rozdělení podle výstupního výkonu.....	5
2.2	Piconet	5
2.3	Zabezpečení Bluetooth	5
2.4	Logické kanály.....	6
2.5	Stavy jednotky Bluetooth	6
2.5.1	Režimy jednotky ve stavu připojení	8
2.6	Protokoly Bluetooth.....	9
2.6.1	Controller stack:.....	9
2.6.2	Host stack:	10
2.7	RFCOMM.....	10
2.7.1	Struktura rámce.....	11
3	Návrh informačních modulů	12
3.1	Blokové schéma.....	12
3.2	Výběr součástí	12
3.2.1	Mikroprocesor (MCU).....	13
3.2.2	Bluetooth modul	14
3.2.3	Displej.....	14
3.3	Schéma zapojení	16
3.3.1	Návrh DPS	16
4	Programování Informačních modulů	18
4.1	ISP/SPI.....	18
4.2	Program pro mikrokontrolér	19
4.2.1	Popis jednotlivých funkcí	19
4.2.2	Sériová komunikace USART.....	20
4.2.3	Chování programu	21
4.3	Výsledné vlastnosti informačního modulu	22
4.3.1	Popis ovládání.....	22
4.3.2	Psaní zprávy:.....	24
4.3.3	Využití kláves při psaní zprávy:	24
4.3.4	DIP Switch.....	25

5	Implementace aplikace pro server.....	26
5.1	QExtSerialPort.....	26
5.1.1	Použité metody z této knihovny.....	26
5.2	Implementované třídy.....	27
5.2.1	Některé metody těchto tříd	27
5.3	Popis aplikace	27
5.3.1	Ovládání.....	28
6	Závěr	30
7	Literatura.....	31

1 Úvod

V dnešní době prožívají bezdrátové technologie obrovský boom. Trendem je odstranit používání překážejících kabelů tam, kde je to technologicky možné. To je dnes v podstatě všude, kde není třeba přenášet elektrické napájení. Bezdrátové technologie se používají pro přenos dat, vytváření lokálních počítačových sítí a sdílení internetu, případně pro přenos zvuku nebo obrazu. Bez podpory nějaké bezdrátové technologie se již neobejde žádné mobilní zařízení, jako jsou mobilní telefony, tablety, notebooky a netbooky.

V této práci je použita bezdrátová technologie Bluetooth, která byla jedna z prvních. Její počátky se datují již od roku 1994. Od této doby však ušla dlouhou cestu a její dnešní podoba je technologicky o mnoho dále. Práce si klade za cíl návrh a realizaci dvou Informačních modulů. Tyto zařízení budou komunikovat se serverem, běžícím na počítači. Budou si vyměňovat krátké textové zprávy, právě pomocí Bluetooth technologie. Z tohoto důvodu je na každém zařízení umístěn jeden Bluetooth modul, pomocí kterého bude komunikovat s Bluetooth transceiverem, umístěným na serveru. Interakci s uživatelem zajišťuje klávesnice, pomocí které se celé zařízení ovládá, a píšou zprávy. Ty budou zobrazeny na jednoduchém znakovém displeji. Řízení celého modulu bude zajištěno pomocí 8-bitového mikrokontroléru. Bude kladen důraz na snadné ovládání a malé rozměry výsledného zařízení. Serverová aplikace umožňuje příjem zpráv od více informačních modulů a zaslání zprávy na všechny připojené zařízení nebo jen na jedno vybrané.

V kapitole 2 bude popsána technologie Bluetooth. Bude zde zmíněn její historický vývoj, vysvětlen princip, na kterém funguje, popsáno zabezpečení a jednotlivé protokoly, včetně RFCOMM, který je u této práce stěžejní.

V kapitole 3 bude znázorněn hardwarový návrh Informačního modulu. Bude zde zobrazeno blokové schéma znázorňující propojení důležitých součástí, které zde také budou popsány. Na konci této kapitoly se nachází schéma desky plošného spoje, společně s jejím popisem a vysvětlen průběh výroby samotné DPS.

Kapitola 4 bude věnována vývoji firmwaru pro informační modul. Budou zde popsány jak použité prostředky, tak samotný program. V této kapitole bude také popsáno výsledné chování zařízení a popsáno jeho ovládání.

V 5. kapitole je popsán vývoj aplikace určené pro server. Bude zde zmíněna použitá externí knihovna qextserialport. Popsány budou také třídy, vytvořené při tvorbě aplikace. Ke konci kapitoly budou umístěny obrazovky z aplikace a popsáno její ovládání a schopnosti.

Výsledky této práce a další možné rozšíření budou shrnuty v závěrečné kapitole 7.

2 Bluetooth

Bluetooth je bezdrátová komunikační technologie, která slouží pro přenos dat na krátké vzdálenosti mezi dvěma nebo i více zřízenými, jako jsou například počítače, mobilní telefony, tiskárny apod. [1]. Mezi těmito zařízeními vytváří osobní síť neboli PAN (Personal Area Network) a pracuje ve frekvenčním pásmu pohybujícím se kolem 2,4 Ghz [1].

Název Bluetooth pochází z anglického překladu jména Dánského krále, kterému se říkalo král Modrozub (Bluetooth) a vládl v 10. století. Využil svých diplomatických schopností, aby válčící kmeny přistoupily k diskuzi a ukončily tím rozepře, čímž se zasloužil o sjednocení Dánska s Norskem.

Technologii Bluetooth vyvinuli Jaap Haartsen a Sven Mattisson, pracující pro společnost Ericsson roku 1994 a byla původně zamýšlena jako bezdrátová alternativa RS-232. V roce 1998 vznikla skupina SIG (Bluetooth Special Interest Group), která se zabývá definováním nových standardů pro Bluetooth. Dnes je v ní 9 velkých společností (Ericsson, 3Com, IBM, Intel, Lucent Technologies, Microsoft, Motorola, Nokia, Toshiba) a přes 13000 členských společností.

Jedním z požadavků při návrhu bylo zajištění celosvětové kompatibility pásma. Nejvhodnější pásmo s těmito požadavky je ISM (Industrial Scientific Medical band). Je bezlicenční a volné pro kterýkoliv rádiový systém, celá šířka pásma je 2,400 – 2,4835 Ghz. Pro potlačení interference s dalšími signály pracujícími na ISM, používá Bluetooth metodu frekvenčních skoků (frequency hopping) s nominální rychlostí 1600 skoků/s.

2.1 Verze Bluetooth, podle historického vývoje

Bluetooth v1.0 a v1.0B

- Tyto verze měly mnoho problémů.
- Firmy měly velké potíže vyrobit kompatibilní produkty s touto verzí.

Bluetooth v1.1

- Opraveno mnoho chyb z předchozí verze.
- Přidána podpora pro nešifrované kanály.

Bluetooth v1.2

- Je zpětně kompatibilní s verzí 1.1
- Rychlejší hledání zařízení a připojování.
- Maximální rychlost přenosu dat je 1 Mb/s.

Bluetooth v2.0 + EDR

- Vydáno roku 2004, zpětně kompatibilní s verzí 1.2
- Hlavním rozdílem je použití EDR (Enhanced Data Rate) pro rychlejší přenos dat.
- Maximální rychlost přenosu dat se tím zvýšila na 3 Mb/s.

Bluetooth v2.1 + EDR

- Opět zpětně kompatibilní se staršími verzemi.
- Hlavní novinka je SSP (Secure Simple Pairing), která vylepšuje párovací schopnosti Bluetooth zařízení a zároveň zvyšuje zabezpečení.

Bluetooth v3.0 + HS

- Opět zpětně kompatibilní se staršími verzemi.

- Rychlost přenosu dat je teoreticky až 24 Mb/s, ovšem v tomto případě se k samotnému přenosu dat se využívá wi-fi a bluetooth je použito pouze pro propojení zařízení.
- Vyšší stabilita a menší energetická náročnost.
- Existuje i verze bez HS (High speed), která je pomalejší a nevyužívá wi-fi pro přenos dat.

Bluetooth v4.0

- Dnes nejnovější verze, schválena v dubnu 2010.
- Přináší větší dosah a menší energetickou náročnost.
- Obsahuje protokoly Classic Bluetooth, Bluetooth high speed a Bluetooth low energy.

2.1.1 Rozdělení podle výstupního výkonu

Class 1 – maximální výkon 100 mW (20 dBm), dosah přibližně 100m

Class 2 – maximální výkon 2,5 mW (4 dBm), dosah přibližně 10m

Class 3 – maximální výkon 1 mW (0 dBm), dosah přibližně 1m

2.2 Piconet

Zařízení bluetooth, které jsou v dosahu, mohou vytvořit spojení mezi dvěma body (Point-to-Point), případně mezi více body (Point-to-Multipoint). Dvě nebo více zařízení sdílející stejný kanál tvoří buňku zvanou Piconet. Několik takových buněk se může spojit a vytvořit uskupení Scatternet a umožnit tak pružnější komunikaci. Pro řízení provozu se jedno zařízení stane řídícím (Master) a ostatní zařízení mu budou podřízenými (Slaves). Podle specifikace může s Masterem aktivně komunikovat až 7 zařízení Slave. Pro přenos hlasu a jiných časově kritických dat jsou určeny synchronní spojově orientovaná spojení SCO (Synchronous Connection Oriented link). Pro ostatní přenosy je určeno asynchronní bezspojově orientované spojení ACL (Asynchronous Connection-Less link).

2.3 Zabezpečení Bluetooth

U Bluetooth přenosů je zabezpečení velice důležitým prvkem [6]. Mezi nejoblíbenější způsoby zneužití technologie Bluetooth patří například:

Bluejacking - zasílání nevyžádané zprávy (spam), všem zařízením v dosahu. Pokud si uživatel neuvědomí, co daná zpráva je, může si zařadit odesilatele zprávy do ověřených kontaktů a ten se k němu poté bude moct kdykoliv připojit. Kvůli malému dosahu Bluetooth se využívá v místech s velkou hustotou lidí, jako například obchodní centra.

Bluebugging – ten už je větším problémem, jelikož umožní hackerovi vzdálený přístup na telefon uživatele a používání jeho funkcí, včetně vytáčení hovorů a posílání sms zpráv. Přitom uživatel si vůbec nemusí uvědomit, že se něco takového děje.

Car Whisperer- jedná se o software, který umožňuje hackerům posílat a přijímat audio z autorádií se zapnutým Bluetooth.

Jelikož je Bluetooth stále poměrně nová a neověřená technologie, neustále vznikají nové možnosti zneužití. Na druhou stranu však výrobci neustále vydávají nové verze firmware, které by to měly vyřešit.

K zabezpečení se používá několik nástrojů. Nejrozšířenější ochrana je pomocí spárování. Kdy musí být zařízení, které chtějí provádět jakýkoliv přenos dat předem spárována. Je to metoda, kde musí uživatelé na obou zařízeních zadat stejný, libovolně zvolený kód (PIN) a poté se uloží adresa druhého zařízení do seznamu bezpečných spojení. Následně už mohou dané zařízení spolu kdykoliv komunikovat bez omezení.

2.4 Logické kanály

U Bluetooth komunikace se nachází pět logických kanálů [2]. Dva slouží pro řízení a správu spojení, tři pro přenos dat. Mezi tyto kanály patří:

Kanál řízení spojení LC (Link Control)

Je mapován do hlavičky paketů, přenáší řídicí informace nízké úrovně, jako jsou například informace o doručení paketu ARQ, řízení toku FLOW a charakter uživatelských informací v informačním poli. Je přenášen ve všech paketech, kromě ID, který nemá hlavičku.

Kanál správy spojení LM (Link Manager)

Přenáší informace, které si vyměňují správci spojení řídicí a podřízené jednotky. Obvykle je pro tento kanál používán paket DM. Je označen v hlavičce informačního pole položkou L_CH.

Kanál asynchronních dat UA (User Asynchronous data)

Přenáší asynchronní data vyšší vrstvy L2CAP. Využívá pakety provozního kanálu ACL.

Kanál izochronních dat UI (User Isochronous data)

Je podporován vyšší vrstvou správným časováním doby vysílání jednotlivých paketů. Pro ty platí na úrovni vrstvy baseband stejné pravidla jako pro pakety kanálu UA.

Kanál synchronních dat US (User Synchronous data)

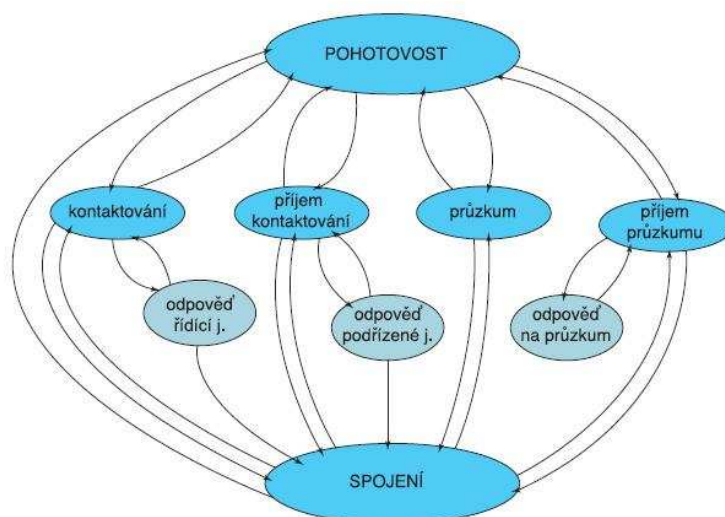
Přenáší synchronní data. Je přenášen pakety provozního kanálu SCO.

2.5 Stavby jednotky Bluetooth

Výchozím stavem je stav pohotovosti (StandBy), kde fungují pouze vnitřní hodiny a jednotka má minimální spotřebu energie. Z tohoto stavu může přejít do jednoho ze stavů: kontaktování (Page), příjem kontaktování (Page Scan), průzkum (Inquiry) nebo příjem průzkumu (Inquiry Scan) [2], viz obrázek 2-1.

Kontaktování (Page)

Do tohoto stavu přechází řídicí jednotka, pokud si přeje navázat spojení s jinou jednotkou Bluetooth. Princip je podobný jako při prohledávání, ale ID-paket obsahuje navíc přístupový kód zařízení DAC (Device Access Code), který je vypočten z adresy kontaktované jednotky. Z té se také řídicí jednotka snaží odhadnout frekvenci, na které bude daná jednotka přijímat. Tento proces bude přesnější a rychlejší, pokud je známa frekvence vnitřních hodin. Frekvenční skoková sekvence kontaktování



2-1: Schéma provozních stavů jednotky Bluetooth [2]

obsahuje 32 frekvencí a je rozdělena do dvou skupin (trains) po šestnácti. Sekvence A obsahuje frekvence z okolí očekávané frekvence a B obsahuje ty ostatní. Pokud je odhad stavu hodin podřízené jednotky v rozsahu $7 \cdot 1,28s$ až $8 \cdot 1,28s$, spustí se sekvence A, jinak se spustí sekvence B. V prvním případě zabere kontaktování maximálně $1,28s$, ve druhém $2,56s$.

Přijem kontaktování (Page Scan)

V tomto stavu podřízená jednotka očekává ID paket s příslušným přístupovým kódem zařízení DAC. Při vstupu do tohoto stavu je ze sekvence pro příjem kontaktování zvolena frekvence, na které jednotka naslouchá určitou dobu. Tato doba by měla být delší, než čas potřebný na prozkoumání šestnácti frekvencí. V případě zadání této doby nulové, bude jednotka naslouchat neustále.

Odpověď na kontaktování (Page response)

Do tohoto stavu jednotka přejde při příjmu kontaktního ID paketu. Podřízená jednotka odešle ID paket se svým přístupovým kódem DAC na stejné frekvenci v následujícím časovém úseku, přesně $625\mu s$ po přijetí kontaktního ID paketu. Řídící jednotka poté taky přejde do stavu odpověď na kontaktování. Nyní zná řídící jednotka frekvenci, na které podřízená jednotka přijímala. Dále pošle řídící jednotka pomocí paketu FHS informace o hodinách řídící jednotky na frekvenci, která byla zjištěna z předchozí odpovědi s použitím přístupového kódu DAC podřízené jednotky. Podřízená jednotka potvrdí příjem paketem ID a použije přijatý FHS paket, ke zjištění přístupového kódu sítě piconet, která byla právě vytvořena, nebo do které podřízená jednotka právě vstoupila a výpočte posuv vnitřních hodin, pro správnou synchronizaci s frekvenční skokovou sekvencí řídící jednotky. Dále řídící jednotka odešle paket POLL, který je již adresován pomocí adresy aktivního člena a je opatřen přístupovým kódem pro danou síť piconet na frekvenci skokové sekvence pro fyzický kanál. Podřízená jednotka na to musí odpovědět jakýmkoliv paketem. Pokud všechno proběhne správně, přejdou obě jednotky do stavu připojení. Poté si jednotky začnou vyměňovat informace pomocí protokolu pro správu spojení LMP (Link Manager Protocol), za účelem vytvoření spojení.

Průzkum (Inquiry)

V tomto stavu jednotka prozkoumává okolí a sbírá adresy dalších zařízení v dosahu. Poté je možné s jakoukoliv z nich navázat spojení přes stav kontaktování. Zkoumající jednotka vysílá průzkumnou zprávu, prostřednictvím paketu ID, na různých přeskokových frekvencích. Paket je možné vysílat

dvakrát ve stejnou dobu na různé přeskokové frekvenci. Podle toho, jestli je paket určen všem nebo jen vybraným jednotkám, obsahuje buď obecný přístupový kód GIAC (General Inquiry Access Code), nebo specializovaný přístupový kód DIAC (Dedicated Inquiry Access Code). Opakovaně vysílá na šestnácti různých frekvencích, které tvoří průzkumnou skokovou frekvenci (train). Ty jsou dvě – A a B. Aby byly zachyceny odpovědi od všech jednotek, je potřeba každou sekvenci projít 256-krát (v případě, že nedochází k chybám). Celkový čas, potřebný na vykonání průzkumu je minimálně 10,24s. Přítomnost provozních kanálů SCO může čas prodloužit. Procedura může být ukončena i dříve, pokud bylo zachyceno požadované množství odpovědí.

Příjem průzkumu (Inquiry Scan)

Do tohoto stavu zařízení periodicky vstupuje, pokud umožňuje identifikaci. Jednotka naslouchá na jedné vybrané frekvenci, aby mohla z průzkumné skokové sekvence zachytit kód odvozený z její BD-adresy. V tomto stavu zůstává dostatečně dlouho, aby zachytila průzkumnou zprávu, která jí patří.

Odpověď na průzkum (Inquiry Response)

Pokud zařízení ve stavu příjem průzkumu, přijme průzkumnou zprávu, musí být zpět odeslána odpověď s adresou jednotky a hodnotou vnitřních hodin. Zpráva není odeslána hned, ale čeká se náhodný počet časových úseků. To zabraňuje případné kolizi s ostatními zařízeními v dosahu, které přijaly stejnou průzkumnou zprávu. Odpověď je poslána v FHS paketu, který obsahuje všechny potřebné informace pro její kontaktování. Po odeslání zůstává ve stavu příjem průzkumu a nečeká na potvrzení vyslaného paketu.

2.5.1 Režimy jednotky ve stavu připojení

Aktivní režim (Active mode)

Jednotka v aktivním režimu obsadí daný kanál. Řídící jednotka plánuje jednosměrný nebo dvousměrný přenos dat, podle požadavků podřízených jednotek a pravidelné přenosy pro udržení synchronizace podřízených jednotek s fyzickým kanálem. Aktivní jednotky naslouchají v časových intervalech určených pro provoz od řídící jednotky, zda pro ně není určen některý její paket. Pokud ne, nemusí naslouchat až do dalšího časového intervalu, vyhrazeného ve směru od řídící jednotky. Počet časových úseků se určí, pomocí hodnoty TYPE v hlavičce paketu.

Režim Sniff (Sniff mode)

V tomto režimu omezuje podřízená jednotka režim naslouchání na krátké časové intervaly, podle aktivity řídící jednotky. To sníží spotřebu energie u podřízené jednotky. Když se podílí na provozním kanálu ACL, musí naslouchat v každém čase, ve kterém začíná přenos od řídící jednotky. V tomto režimu je počet úseků, kdy může řídící jednotka komunikovat s podřízenou omezeno tak, že řídící jednotka může zahájit přenos pouze v určitých časových úsecích. Ty jsou rozděleny do pravidelných intervalů T_{sniff} . Tento režim je řízen pomocí protokolu LMP.

Režim přidržení (Hold mode)

Během připojení může být provozní kanál ACL uveden do režimu přidržení. Taková jednotka dočasně nepodporuje pakety na provozním kanále ACL (SCO je však stále podporováno). Uvolněná kapacita může být použita pro jiný účel, jako například průzkum, kontaktování nebo účast v jiné síti piconet. Případně se může brát jako režim s nižší spotřebou energie. Před vstupem do režimu

přidržení, se řídící a podřízená jednotka dohodnou, jak dlouho bude trvat. Po jeho skončení se jednotka přepne opět do aktivního režimu, připojí se na fyzický kanál a čeká na instrukce od řídící jednotky.

Režim parkování (Park mode)

Je to režim s nejnižší spotřebou energie podřízené jednotky. Používá se také jako režim, který má možnost připojení více než sedmi zařízení k jedné řídící jednotce. Při vstupu do tohoto režimu odevzdá podřízená jednotka adresu aktivního člena AM_ADDR a jsou jí přiděleny dvě osmibitové adresy. Adresa zaparkovaného člena PM_ADDR (Parked Member Address), kterou používá řídící jednotka pro změnu režimu a žádosti o přístup AR_ADDR (Access Request Address), používanou podřízenou jednotkou na žádost o změnu režimu. Řídící jednotka v pravidelných intervalech naslouchá na fyzickém kanále, kvůli synchronizaci a sledování žádosti o změnu režimu.

2.6 Protokoly Bluetooth

Bluetooth používá řadu různých protokolů [4]. Ty hlavní jsou definovány organizací Bluetooth SIG. Další byly převzaty z jiných skupin standardů. Zásobník bluetooth protokolu je rozdělen na dvě části: zásobník kontroléru a zásobník hosta.

Zásobník kontroléru (controller stack) je obvykle na levných zařízeních obsahujících mikroprocesor a bluetooth vysílač.

Hostitelský zásobník (host stack) většinou se nachází na zařízeních s operačním systémem.

2.6.1 Controller stack:

ACL - Asynchronous Connection-oriented [Logical Transport]

Používá se u spojení pro přenos obecných datových paketů. Připojení musí být přesně nastaveno a přijato mezi dvěma zařízeními předtím, než se umožní přenášení paketů. ACL automaticky přeposílá v případě, že pakety nebyly přijaty a potvrzeny.

SCO - Synchronous connection oriented

Používá se u spojení pro přenos hlasových dat. SCO ke své činnosti využívá existující ACL spojení. Pakety u SCO nejsou přeposílány, v případě nedoručení jsou ztraceny.

LMP - Link Management protokol

Používá se pro řízení komunikace mezi dvěma zařízeními. Funkce LMP: správa spojení – funkce pro vytváření spojení, bezpečnost – párování, autentizace, výměna klíčů, šifrování atd., řízení sítě piconet a konfigurace spojení – kontrola kvality spojení, volba paketů atd.

Pakety LMP mají vyšší prioritu než L2CAP, takže nejsou zpomalovány běžným provozem na kanálu. Vlastní LMP paket se skládá ze tří částí: Transaction ID (informace o původu transakce), OpCode (určuje druh zprávy) a obsahu.

HCI - Host/controller interface

Standardizovaná komunikace mezi hostitelským zásobníkem a kontrolérem.

2.6.2 Host stack:

L2CAP – Link Control and Adaptation Protocol

Je používán uvnitř bluetooth rozhraní, posílá síťové pakety na HCI nebo přímo na Link Manager. Komunikuje přes hostitelské ACL spojení. Spojení je navázáno poté, co bylo ACL spojení nastaveno. ACL kanály jsou určeny pomocí identifikátoru kanálu CID (Channel Identifier). Kanály u L2CAP jsou analogií portů u TCP/IP a jsou jednoznačně identifikovány adresou zařízení, ke kterému jsou vedeny. Každý kanál je plně duplexní a má možnost specifikovat QoS pro oba směry přenosu. Pakety L2CAP nemají žádný mechanismus pro kontrolu a opravu chyb, tyto funkce již zajišťují nižší vrstvy.

BNEP – Bluetooth Network Emulation Protocol

Je používán na doručování paketů v PAN (Personal Area Network) profilech. BNEP umožňuje podobné funkce jako SNAP ve Wireless LAN.

RFCOMM – Radio Frequency Communication

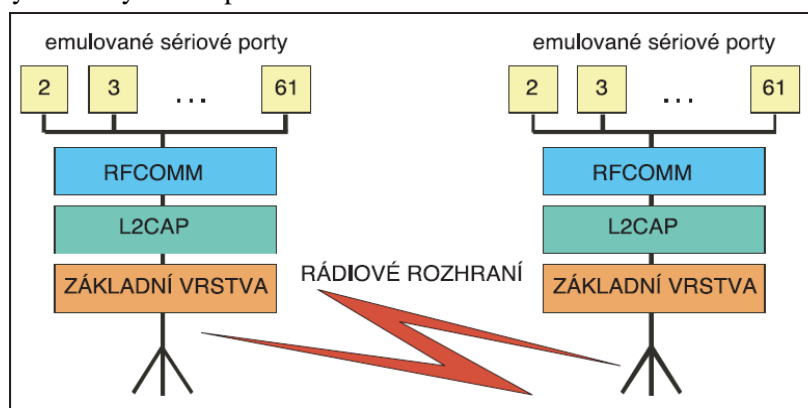
Sada přenosových protokolů, založených na L2CAP protokolech. Využívá emulovaný sériový port, přes který probíhá komunikace.

SDP - Service discovery protokol

Je používán, aby umožnil zařízením zjistit, které služby navzájem podporují a jaké parametry se mají použít pro připojení k nim.

2.7 RFCOMM

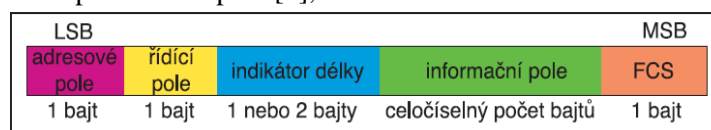
Protokol RFCOMM v této práci popíšeme podrobněji, jelikož je použit pro přenos dat mezi informačním modulem a serverem. Slouží jako nosný protokol pro různé aplikační protokoly. Emuluje sériové rozhraní RS232, definovaným ve specifikaci ETSI TS 07.10 s drobnými změnami, které vyplývají z vlastností technologie Bluetooth [7]. Zařízení tohoto typu se po spárování přiřadí neobsazené číslo COM portu, přes které se poté se zařízením komunikuje, jako by se jednalo o klasický sériový port. RFCOMM podporuje až 60 nezávislých spojení. V TS 07.10 jsou možné tři pracovní režimy: základní, rozšířený bez oprav chyb a rozšířený s opravou chyb. RFCOMM byl odvozen od základního režimu, ale nepoužívá princip transparentnosti a návěští začátku a konce rámce. Zjednodušený vrstevný model protokolu RFCOMM můžete vidět na obrázku 2-1.



Obrázek 2-2: Zjednodušený vrstevný model RFCOMM [2]

2.7.1 Struktura rámce

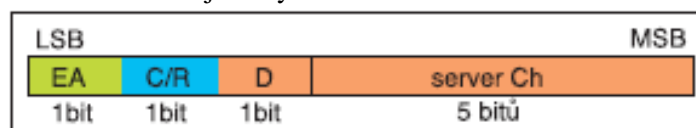
Rámec protokolu RFCOMM je složen z adresového a řídicího pole, identifikátoru délky, informačního pole a zabezpečovacího pole [2], viz obrázek 2-2.



Obrázek 2-3: Formát rámce RFCOMM [2]

Adresové pole

Má velikost 1 byte. Obsahuje pole EA, C/R, D a kanál serveru, viz obrázek 2-3. Kombinace D (Direction bit) a kanálu serveru (Server channel) udává identifikátor datového kanálu DLCI (Data Link Connection Identifier). Serverové aplikace na zařízeních nezahajujících spojení mají DLCI 2, 4, 6, ..., 60 (sudá čísla). Serverové aplikace na zařízeních zahajujících spojení mají DLCI 3, 5, 7, ..., 61 (lichá čísla). Bit C/R (Command/Response) rozlišuje, zda se jedná o příkaz nebo odpověď. Jeho hodnota také závisí na tom, zda je jednotka iniciátorem spojení. Bit EA (Extension Address) označuje konec adresového pole. U RFCOMM je vždy rovno 1.



Obrázek 2-4: Formát adresového pole [2]

Řídící pole

Definuje typ rámce. Existuje pět typů rámců:

- SABM (Set Asynchronous Balanced Mode) - příkaz pro zahájení komunikace na určeném DLCI.
- UA (Unnumbered Acknowledge-ment) - pozitivní odpověď na příkaz.
- DM (Disconnected Mode) - negativní odpověď při pokusu o zahájení komunikace.
- DISC (Disconnect) - příkaz pro ukončení komunikace na určitém DLCI,
- UIH (Unnumbered Information with Header check) – používá se pro řídicí příkazy a přenos dat.

Identifikátor délky

Začíná bitem EA. Pokud je nastaven na 1, potom nižších 7 bitů udává délku a pole je velké 1 byte. Pokud je EA v 0, tak je délka udána v následujících 15 bitech a celé pole má velikost 2 byty. Základní délka RFCOMM rámce je 32 bytů. Maximální délka je 32767bytů.

Informační pole (data)

Vlastní přenášená data. Musí být vždy celočíselný počet bytů, až do délky 32767. Délkový limit je nastaven MTU (Maximum Transmission Unit) v L2CAP paketech. Takže pokud má systém menší L2CAP MTU, bude velikost RFCOMM dat také omezena.

FCS (Frame Check Sequence)

Slouží k zabezpečení rámce pomocí cyklického kódu. Jeho hodnota se počítá pro různé typy rámců. Pro SABM, DISC, UA, a DM rámce je počítán z adresového a řídicího pole a indikátoru délky. Pro UIH rámce, se počítá z adresového a řídicího pole.

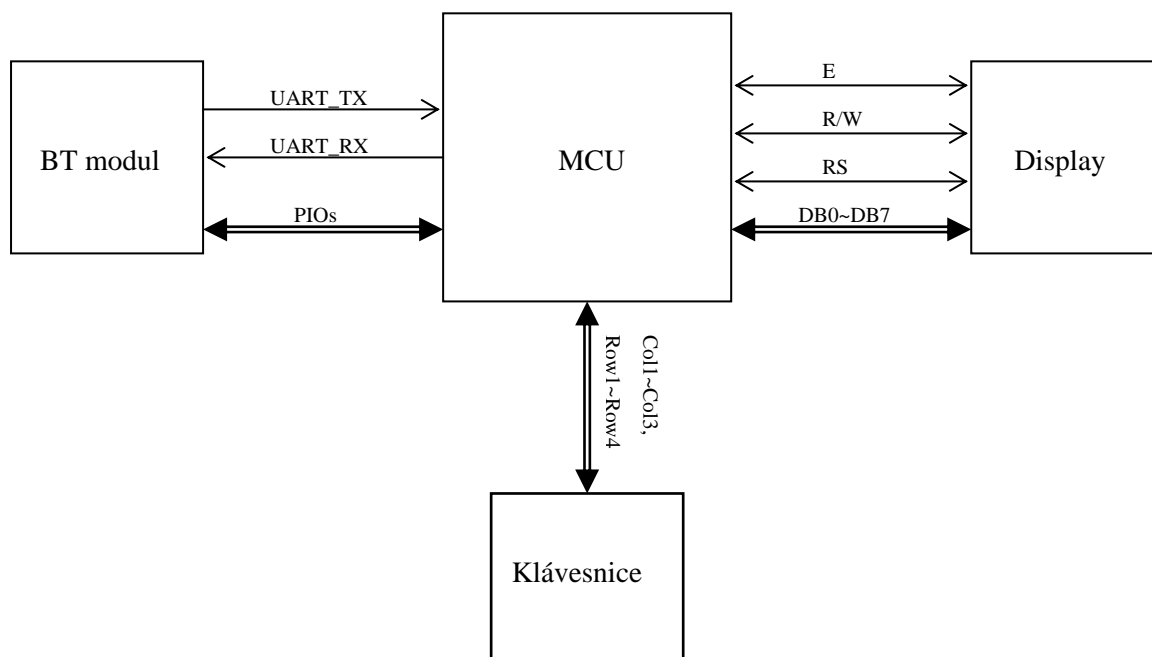
3 Návrh informačních modulů

V rámci této bakalářské práce byly vytvořeny 2 Informační moduly pro Bluetooth komunikaci se serverem, běžícím na PC. V této kapitole je popsán a zobrazen postupný vývoj těchto modulů a jejich realizace. Nejprve návrh, potom výběr jednotlivých součástí i s jejich stručným popisem a nakonec popis schématu a návrh DPS.

3.1 Blokové schéma

Hlavní částí návrhu je 8-bitový mikrokontrolér, který bude komunikovat s Bluetooth modulem pomocí sériové smyčky UART. Mikrokontrolér bude také číst nebo nastavovat jeho vstupní/výstupní porty. Komunikace uživatele s mikrokontrolérem je vyřešena pomocí klávesnice a displeje. Klávesnice bude pouze jednoduchá s 12 tlačítky (4 řádky, 3 sloupce) a uživatel pomocí ní bude zadávat zprávy stejným způsobem, jako na klasickém mobilním telefonu. Displej bude sloužit pro zobrazování přijatých zpráv, psaní zprávy a zobrazení stavu (připojen/odpojen), pro jednoduchost je zvolen model s paralelním zápisem dat.

Propojení jednotlivých částí mezi sebou je vidět na obrázku 3-1.



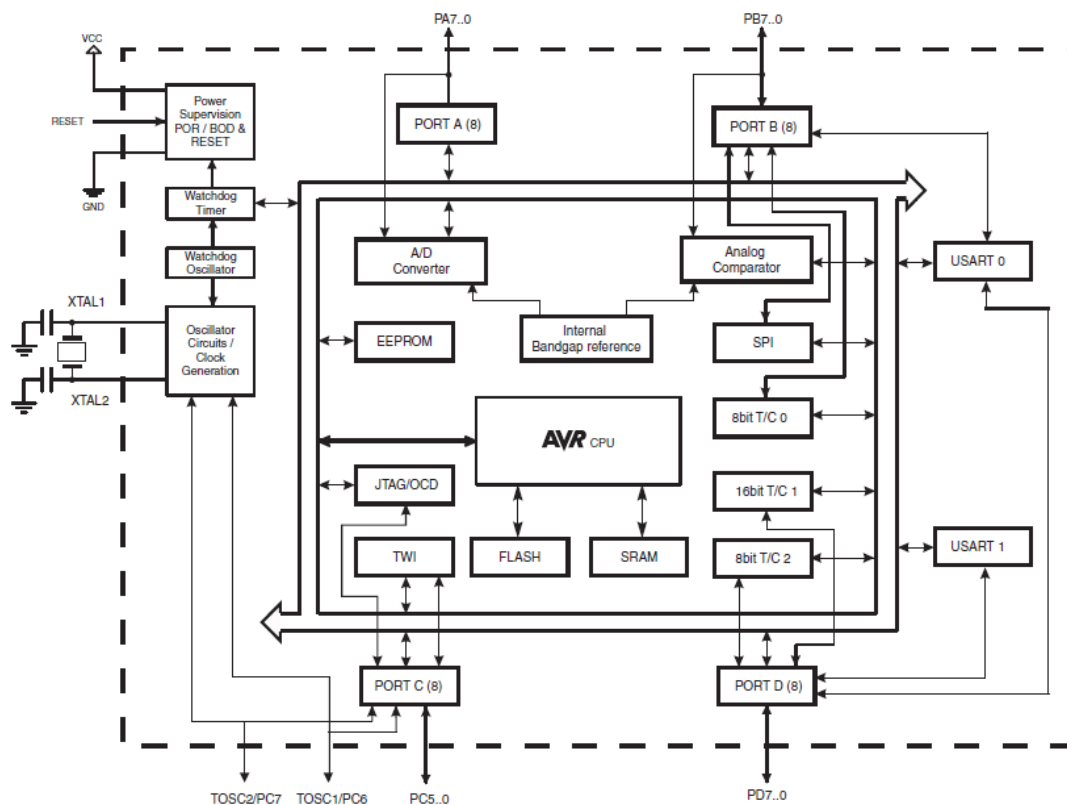
Obrázek 3-1: Blokové schéma modulu

3.2 Výběr součástí

Po návrhu blokového schématu, bylo potřeba vybrat vhodné typy 4 základních součástí, které jsou znázorněny na obrázku 3.1. Všechny součásti byly s ohledem na schopnost pracovat na napětí kolem 3V, aby mohl být celý modul jednoduše napájen z klasických, dobře dostupných baterií a nebyla narušena mobilita modulu, například napájením ze zdroje.

3.2.1 Mikroprocesor (MCU)

Mikroprocesor firmy Atmel byl zvolen z důvodu dostupnosti, jednoduché manipulace a možnosti zapůjčení programátoru Asix PRESTO. Byl vybrán typ Atmel ATMEGA644PA-MU řady AVR [8], z důvodů ceny a s velkou rezervou dostačující rychlosti a velikosti paměti programu. Konkrétně typ v pouzdru TFQP44, pro SMD montáž, které je vhodný kompromis mezi velikostí čipu a složitosti pájení na DPS (Deska Plošného Spoje). Blokové schéma zvoleného mikrokontroléru můžete vidět na obrázku 3-2.



Obrázek 3-2 - Blokové schéma Atmega644PA [8]

Klíčové vlastnosti:

- Programování vnitřní flash paměti přes ISP (In-System Programming), pomocí SPI nebo JTAG
- 8-bitová RISC architektura (131 instrukcí, většina jedno cyklových)
- Rychlost až 20 MIPS při 20Mhz
- 64KB vnitřní (In-System Self-programmable) Flash paměti pro program
- 2048B paměti EEPROM
- 4KB vnitřní SRAM paměti
- Počet zápisových/mazacích cyklů: 10 000 Flash/ 100 000 EEPROM
- Dva 8-bitové čítače a jeden 16-bitový čítač
- Dvě programovatelné sériové linky USART
- Interní a externí zdroje přerušení
- Interní RC oscilátor, případně možnost použití externího krystalu pro větší rychlost
- 32 programovatelných vstupů/výstupů
- Pracovní napájení 1,8V-5V (0-4Mhz od 1.8V, 0-10Mhz od 2.7V, 0-20Mhz od 4.5V)

3.2.2 Bluetooth modul

Byl vybrán Bluetooth modul RN-41 od firmy Roving networks [10]. Zejména z důvodů malé velikosti a obsazení integrované antény na modulu.

Je typu class 1, která zaručuje dosah 100m, ale má snížená napájecí nároky (250uA deep sleep, <10mA sniff mode, 30mA connected). Podporuje Bluetooth v2.1, maximální rychlost přenosu dat je však omezena použitím protokolu RFCOMM, s čímž souvisí zachování rychlosti přenosu, které platí pro komunikaci přes sériový port COM. Pracuje v profilu SPP, za použití protokolů RFCOMM a L2CAP. Komunikace s modulem a jeho nastavování může probíhat přes sériovou linku UART (SPP nebo HCI), případně přes USB (pouze HCI). Modul pracuje s napětím od 3,0V do 3,6V.

Připojení k modulu:

Aby mohl server (počítač) s modulem komunikovat je třeba, aby s ním spárován, tuto proceduru je nutné provést pouze před první komunikací mezi daným serverem a jednotlivým Bluetooth modulem. Jakmile bude spárování jednou provedeno, modul se přidá do systému jako nové zařízení a poté již je možno s ním kdykoliv komunikovat.

Postup spárování ve Windows 7:

1. Zapněte Bluetooth na vašem počítači a také zapněte Bluetooth modul RN-41
2. Klikněte na „Zařízení Bluetooth“ a zvolte „přidat zařízení“
3. Systém nyní prohledá Bluetooth zařízení v dosahu, jedním z nich by měl být náš modul. Defaultně má jméno „Firefly-ABCD“, kde ABCD jsou poslední 4 znaky z jeho MAC adresy. Pokud takovéto zařízení nebylo nalezeno, je možné, že bylo přejmenováno nebo je chyba v komunikaci ze strany serveru nebo modulu.
4. Vyberte zařízení a stiskněte tlačítko „další“
5. Z nabídky zvolte „Zadat párovací kód zařízení“ a klikněte na tlačítko „další“
6. Zadejte defaultní kód 1234 a klikněte na další. Tímto se Bluetooth modul přidá mezi ověřené Bluetooth zařízení a párování je dokončeno.

Číslo portu COM, na kterém bylo zařízení po spárování přiřazeno, zjistíte kliknutím na Bluetooth zařízení – Otevřít nastavení – Porty COM. Pro komunikaci se používají porty, kde je u jména zařízení na konci napsáno ‘SPP’.

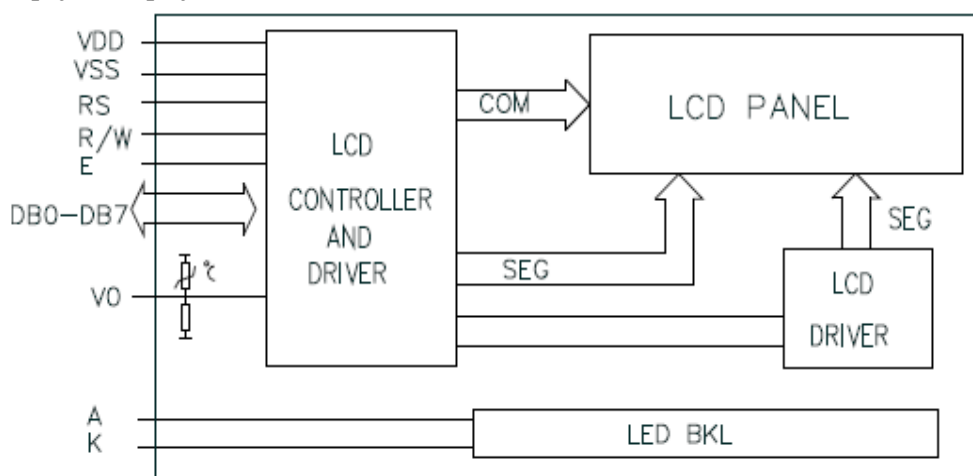
Konfigurace modulu:

RN-41 se dá konfigurovat vzdáleně nebo přímo přes sériovou smyčku. Nejprve se musíme dostat do konfiguračního módu. To uděláme zasláním „\$\$\$“. Od modulu přijde odpověď „CMD“. V základu je modul nastaven tak, že do konfiguračního módu se lze dostat nejpozději minutu po zapnutí modulu. To můžeme změnit příkazem ST,<číslo>, kde číslo v rozsahu (0,255) udává čas v sekundách, 0 znamená zákaz konfigurace a 255 znamená povolení konfigurace neustále. Další příkazy, které je možné v tomto módu zadávat, najdete v manuálu [11]. V případě zadání správného formátu příkazu modul odpoví „AOK“, jinak „ERR“. Z konfiguračního módu se dostanete zasláním ---<CR>.

3.2.3 Displej

Byl vybrán display NHD-0216HZ-FSW-FBW-3V3C od firmy Newhaven [9]. Tento displej je pouze znakový, umí zobrazit 16x2 znaků, což je takový vhodný kompromis mezi jeho velikostí a

použitelnosti při psaní a čtení zpráv. Posílání znaků příkazů je zde vyřešeno paralelně, takže odpadnou problémy se synchronizací hodin a podobně. Pracuje s napětím 3,3V to byl také jeden z hlavních důvodů pro výběr právě tohoto modelu. Má bílé podsvícení a rozlišení znaku 8x5 pixelů. Blokové zapojení displeje můžete vidět na obrázku 3-3.



Obrázek 3-3: Blokové schéma displeje [9]

Vstup VDD a VSS slouží pro přivedení napájecího napětí (+3,3V a GND).

RS (Register Select) určuje, zda zadáváme příkaz (RS=0) nebo zda zadáváme znak, který chceme zapsat (RS=1).

R/W (Read/Write) signál, kterým se určuje, zda chceme na displej zapisovat nebo z něj číst.

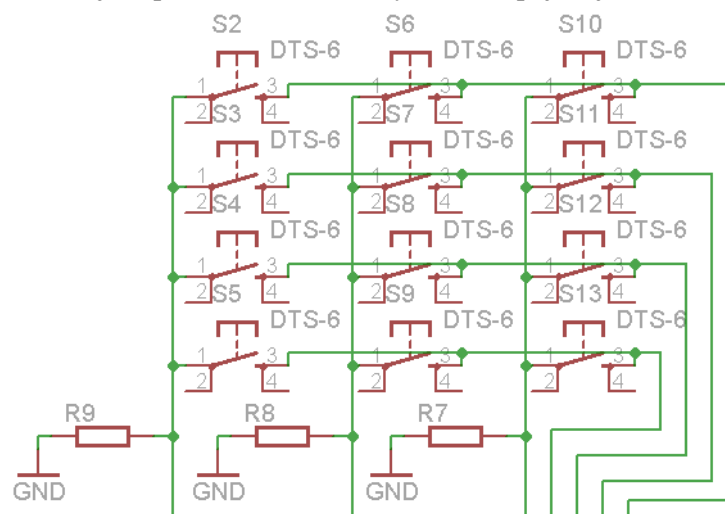
E (Enabled) povoluje zápis/čtení dat DB0-DB7

DB0-DB7 datové vstupy/výstupy, přes které se zapisují znaky, příkazy a zapisuje/čte adresa kursoru.

A, K vstupy pro napájení podsvícení displeje.

Klávesnice

Klávesnice byla nakonec vyřešena umístěním mikrospínačů TACT-64N-F přímo na DPS do matice 3x4. Výhody tohoto řešení je úspora místa a cena. Výsledné zapojení je vidět na obrázku 3-4.



Obrázek 3-4: zapojení klávesnice

Všechny spínače v jednotlivých řadách i ve sloupcích jsou navzájem propojeny. Řady jsou připojeny na výstupy mikrokontroléru a sloupce jsou přivedeny na vstupy mikrokontroléru. Zjištění který z dvanácti spínačů byl stisknut, je řešeno následujícím způsobem: mikrokontrolér posílá cyklicky stav log. 1 na jednotlivé řady (vždy pouze na jednu řadu v daný čas) a poté ze vstupů čte, zda se na nějakém vstupu (řadě), také objevil stav log. 1. Pokud ano, víme ve které řadě a sloupci se nachází stisknutý spínač.

Rezistory R7, R8 a R9 jsou tzv. „stahovací odpory“ a slouží k tomu, aby se vstup mikrokontroléru pro danou řadu, nenacházel v nedefinovaném stavu. V případě, že je stisknutý jakýkoliv spínač v dané řadě, dostane se stav log. 1 (+3,3V) až na vstup mikrokontroléru. Pokud ne, bude vstup přes rezistor uzemněn a bude ve stavu log. 0.

3.3 Schéma zapojení

Schéma zapojení bylo tvořeno ve freeware verzi programu Eagle 5.11.0. Tato verze má omezení v zákazu pro komerční využití a maximální velikost DPS je omezena na 100x80mm, což však pro účely této práce bez problémů dostačuje.

Výsledné schéma zapojení můžete vidět na obrázku nacházejícím se v příloze 1 na konci tohoto dokumentu. Je na něm vidět, že na desce plošného spoje bude mikrokontrolér Atmega644pa a tlačítka klávesnice. Bluetooth modul RN-41, displej a programátor, budou připojeny do konektorů. Konektor SV4 slouží pro připojení programátoru Asix PRESTO a také pro přivedení napájení (SV4-2 +3,3V a SV4-3 GND). SV2 je určen pro připojení Bluetooth modulu Rowing network RN-41. Displej je připojen přes sběrnici J1. Přesné využití všech vstupně/výstupních portů mikrokontroléru a jejich zapojení je uvedeno v tabulce v příloze 1 na konci tohoto dokumentu. Spínače S2 – S13 slouží jako tlačítka pro klávesnici. Ve schématu se dále nachází DIP přepínač, který je přiveden na vstupy Bluetooth modulu a slouží pro jeho nastavování, přesný popis jeho funkcí je pospán dále v podkapitole 4.3.4. Vyhlazovací kondenzátory C3-C7 slouží k odstranění špiček z průběhu vstupního napájení. 10MHz krystal, zapojený přes kondenzátory C1 a C2 proti zemi, je určen jako hodinový vstup pro mikrokontrolér (ten umí pracovat i na frekvenci 20MHz, to by ovšem musel být napájen na 5V a 10Mhz bude pro účely tohoto projektu plně dostačující). Potenciometr R1 slouží k nastavení kontrastu displeje. Rezistory R6, R7 a R8 slouží k uzemnění vstupů mikrokontroléru, v případě neseprnutých spínačů dané řady. Seznam všech použitých součástek je uveden v příloze 1 na konci této práce.

3.3.1 Návrh DPS

Byly vyrobeny dvě dvouvrstvé desky plošného spoje, jejich výsledné rozložení je zobrazeno v příloze 1. Na horní straně DPS je použito tzv. „rozlití mědi“ pro kvalitnější rozvod země (GND). Její rozměry jsou 80mm x65,5mm a na horní polovině jsou vyvrtány 4 díry, které slouží k uchycení Desky plošného spoje s displejem. DPS byly vyrobeny na přístrojích v laboratoři VUT FIT panem Ing. Václavem Šimkem.

Výsledný vzhled modulů můžete vidět na obrázcích v příloze 2. Informační moduly mají rozdílnou barvu, jelikož na DPS jednoho z nich byla vytvořena maska (zelený) a na druhém ne. Na modulu s vytvořenou maskou je bohužel horší kvalita pájených spojů, jelikož při výrobě se maska dostala i na kontaktní plošky a nepodařilo se jí odtud plně odstranit, proto na některých místech

nemohl pájecí cín dokonale přilnout. Ovšem i přes tento problém jsou nakonec obě zařízení plně funkční.

Informačních moduly byly v průběhu vyvíjení napájeny ze zdroje stejnosměrného napětí. Byl ovšem otestován i jejich provoz ze dvou AA baterií v sériovém zapojení (3V). Pro napájení z baterie/akumulátoru o napětí větším než 3,3V by bylo potřeba přidat stabilizátor napětí.

4 Programování Informačních modulů

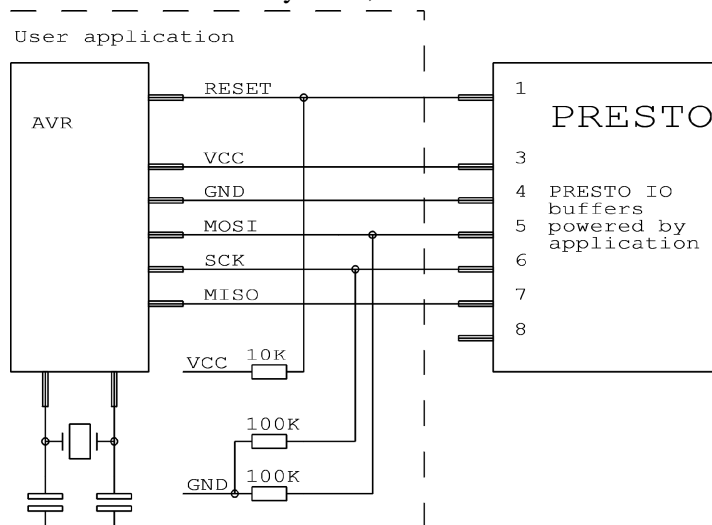
V této kapitole je popsán postup při programování firmware pro informační moduly a také popsáno chování programu. Dále jsou zmíněny jeho výsledné vlastnosti a je popsáno ovládání spolu s obrázky, zobrazujícími stavy displeje a nakonec je popsána funkce jednotlivých tlačítek klávesnice v režimu psaní zprávy.

4.1 ISP/SPI

ISP (In-System Programming) je schopnost některých mikrokontrolérů, být naprogramovány uvnitř obvodu, bez nutnosti vyjmutí čipu a jeho umístění do speciálního programátoru. V této práci byla pro programování mikrokontroléru, použita ISP metoda pomocí SPI.

SPI (Serial Peripheral Interface) je sériové periferní rozhraní [17]. Používá se pro komunikaci mezi řídicími mikroprocesory a ostatními integrovanými obvody. Umožňuje vzájemné propojení dvou či více komunikujících uzlů, kde jeden vystupuje v roli řadiče sběrnice (master), ostatní uzly pracují v režimu slave. Uzel, který pracuje jako master, obsahuje generátor hodinového signálu rozvedený do všech ostatních uzlů, čímž je umožněn synchronní, obousměrný přenos dat. Hodinový signál se rozvádí vodičem označovaným SCK. Dále jsou uzly propojeny vodiči MISO (Master In, Slave Out) a MOSI (Master Out, Slave IN), přes které se přenášejí data.

Na samotné programování mikrokontroléru byl použit programátor PRESTO od firmy ASIX. Jedná se o rychlý programátor, který podporuje programování osazených součástek. Podporuje velké množství součástek. Umožňuje jak programování FLASH paměti, tak i paměti EEPROM. Samotné programování mikrokontroléru Atmega644pa, bylo prováděno pomocí dodávaného programu ASIX UP. Při programování mikrokontrolérů ze série AVR je třeba dávat pozor při nastavování „pojistek“, špatným nastavením je možné celý obvod znehodnotit. Například špatným zvolením zdroje hodin nebo vypnutím RESETu, poté už jej není možné některými programátory znovu naprogramovat. Propojení programátoru presto s mikrokontrolérem řady AVR, můžete vidět na obrázku 4-1.



Obrázek 4-1: propojení programátoru PRESTO s mikrokontrolérem [13]

4.2 Program pro mikrokontrolér

Řídící program pro mikrokontrolér Atmega644PA, byl psán v jazyce C. Konkrétně v programu AVR studio 4 od firmy Atmel s použitím překladače avr-gcc [12].

4.2.1 Popis jednotlivých funkcí

`void USART_Init()`

Provede nastavení sériové linky USART pro komunikaci s Bluetooth modulem.

`void USART_Transmit(unsigned char data)`

Přenos 1 bytu dat (předaných parametrem data) po sériové smyčce.

`void BTcommand(char *string)`

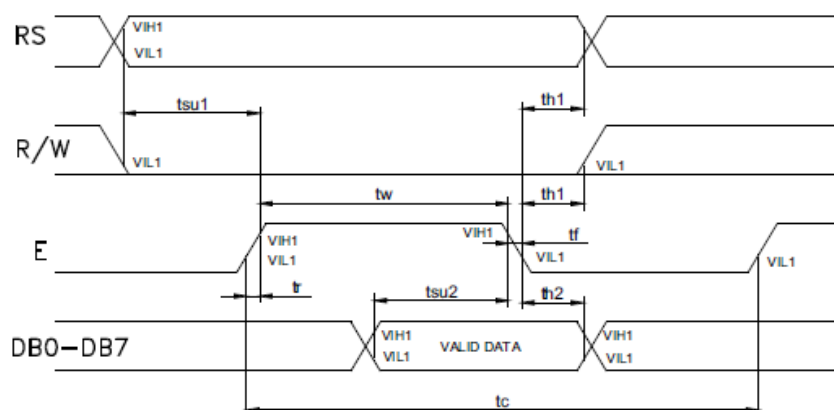
Přepne Bluetooth modul do „command mode“, provede příkaz předaný v parametru string a vrátí modul zpět do „data mode“.

`void display_init(void)`

Provede inicializaci displeje.

`void display_writeCom(unsigned char command)`

Provede příkaz na displeji předaný přes parametr command. Průběh zápisu příkazu na displej je možno vidět na obrázku 4-2. (RS=0).



Obrázek 4-2: průběh zápisu na displej [9]

`void display_writeData(unsigned char data)`

Zapíše znak, jehož ASCII hodnota je předána v parametru data. Průběh zápisu příkazu na displej je možno vidět na obrázku 4-2. (RS=1).

`void display_writeString(char* Str)`

Zapíše řetězec předaný parametrem string na displej.

`unsigned char display_readAddr(void)`

Vrátí adresu (pozici), na které se aktuálně nachází kurzor.

`unsigned char pressed_key()`

Vrací číslo aktuálně stisknuté klávesy (1-12). Pokud není stisknuta žádná klávesa, vrací 0.

`void keyboard_write(void)`

Obsluha klávesnice v režimu psaní zprávy. 1. klávesa odeslání zprávy, 2. klávesa přepínání mezi malými a velkými písmeny, 3. klávesa mazání posledního znaku, 4.-12. klávesa psaní textu.

`void write_charracter(void)`

Volaná právě z funkce keyboard_write při stisknutí kláves 4-12. Obsluhuje psaní zprávy stejným stylem jako na klasickém mobilním telefonu. Tzn. jednou klávesou, můžete například psát

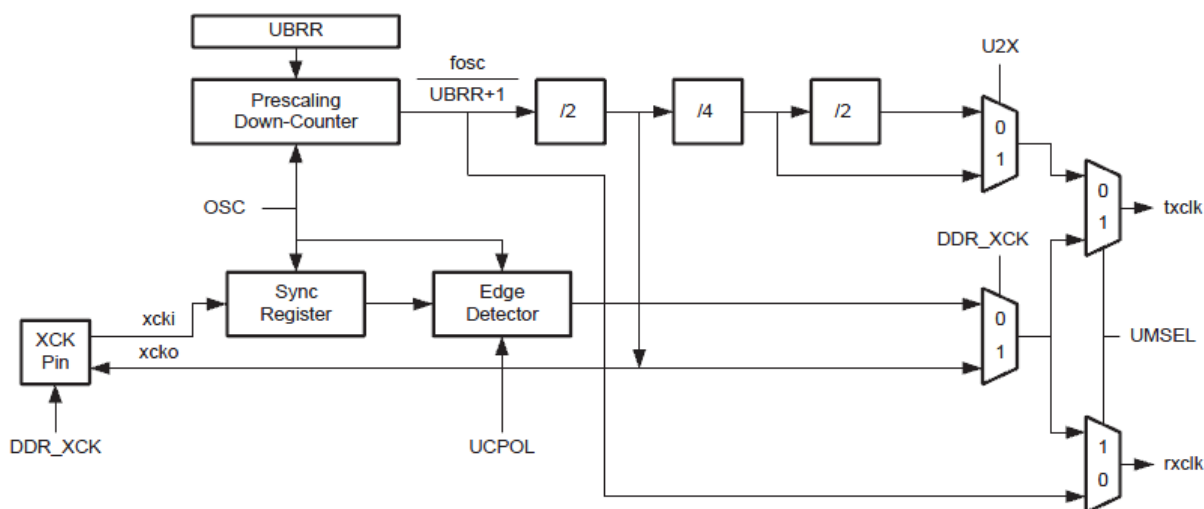
```
void delay buttonUp(void)
```

```
void printPage(unsigned char page, char
```

Informační modul umožňuje přijímat zprávy dlouhé až 160 znaků, vzhledem k tomu, že displej zobrazí najednou pouze 32 znaků, je zde tato funkce pro zobrazení jednotlivých „stránek“. Parametr `page` udává, kolikátá „stránka“ ze zprávy se bude tisknout, `message` je pole s přijatými zprávami a parametr `start` udává od kolikátého znaku je zpráva zobrazována na displeji.

Zobrazí na displeji základní obrazovku. Parametr `newMessages` udává, kolik bylo přijato nových zpráv. Pokud je `newMessages` 255, vytiskne obrazovku se stavem nepřipojeno. Když je 0, vytiskne obrazovku se stavem připojeno. V ostatních případech vytiskne obrazovku s počtem nově přijatých zpráv.

Důležitou částí bylo zprovoznění komunikace mezi Bluetooth modulem a mikrokontrolérem. Ta probíhá přes sériovou smyčku USART [8]. Nejprve je potřeba získat hodinový signál pro sériovou komunikaci z hlavních hodin (krystalu). Postup jak mikrokontrolér získá tento hodinový signál, je vidět na obrázku 4-3.



$$UBRR = \frac{f_{osc}}{16 * BAUD} - 1$$

Z toho vyplývá, že pro BAUD = 57600, je třeba při frekvenci použitého krystalu fosc= 10Mhz nastavit UBRR=10, přesněji UBRR0H = 0 a UBRR0L=10; jelikož registr UBRR je 16-bitový a jedná se o sériovou linku USART0 (Atmega644PA má ještě jednu sériovou linku USART1), abychom se co nejlépe přiblížili požadované hodnotě.

Dále je třeba povolit vysílač a přijímač, to provedeme nastavením bitů RXEN0 a TXEN0 v registru UCSR0B a nakonec nastavit požadovaný frame format v registru UCSR0C. Nyní již máme sériovou linku nastavenou pro komunikaci.

Posílání dat:

```
while ( !( UCSR0A & (1<<UDRE0)) ) ;  
UDR0 = data;
```

Nejprve je třeba ověřit, zda je vysílací buffer prázdný, případně počkat na jeho vyprázdnění. Poté umístění dat na buffer UDR0 jej automaticky odešleme.

Příjem dat:

```
if ( !(UCSRnA & (1<<RXCn)) )  
data = UDR0;
```

Nejprve ověříme, zda jdou na bufferu nějaké data, které můžeme přijmout. Jestli jsou, uložíme si je z bufferu UDR0.

Použité nastavení sériové komunikace USART:

- Baud rate 57,600
- 8 bitů
- Bez Parity
- 1 stop bit
- Flow control vypnut

4.2.3 Chování programu

V hlavním programu je nekonečná smyčka, takže program běhá po zapnutí informačního modulu neustále dokola, až do jeho vypnutí.

Nejprve je testován buffer sériové linky, zda byly přijaty nějaké nové zprávy. Pokud ano, ověří, zda začínají zvoleným bytem pro zprávy, v takovém případě uloží celou zprávu do pole s přijatými zprávami a zobrazí na základní obrazovce počet nových přijatých zpráv. V případě, kdy je přijatý byte ověřením komunikace od serveru, odešle stejný byte serveru zpět, ten má poté ověřeno, že je daný informační modul připojený a vzájemná komunikace funguje. Pokud začínají přijatá data jinak, zahodí se.

Dále se ověřuje, zda je stisknuto nějaké tlačítko. Potom se musí rozlišit, v jakém stavu se informační modul právě nachází a podle toho na stisknuté tlačítko reagovat. V režimu psaní zprávy to může být psaní znaků, mazání znaku, přepínání malá/velká písmena a odeslání zprávy. V režimu čtení zpráv listování v přijaté zprávě nebo přechod na další zprávu/základní obrazovku. Posledním režimem je základní obrazovka, která však rozlišuje tři stavy: připojeno, nepřipojeno a přijaty nové zprávy. V nepřipojeném stavu nejsou žádné reakce na stisk tlačítek, v připojeném stavu je pouze reakce na přechod do psaní nové zprávy a při nových přijatých zprávách je může buď zobrazit, nebo smazat.

Poté je zjišťován stav Bluetooth modulu. V případě změny stavu (připojen - nepřipojen nebo nepřipojen - připojen) je přepsán aktuální stav na hlavní obrazovku a také se změní možnosti, které může informační modul vykonávat.

Nakonec je pouze zařazeno čekání 10ms a inkrementace proměnné `press_delay`, která je po každém stisku kláves nulována. Je zde z důvodu počítání prodlevy mezi vícenásobným stiskem stejné klávesy při psaní zprávy.

Následně provádí celá posloupnost opět znovu od začátku.

4.3 Výsledné vlastnosti informačního modulu

Oba vytvořené informační moduly mají ve finální fázi tyto vlastnosti:

- Rozměry: 80mm x 65,5mm x 25mm
- Váha 81g
- Napájecí napětí 3V, odběr proudu cca. 70mA.
- Displej s aktivním modrým podsvícením, který zobrazí 32 znaků (rozlišení znaku 8x5 pixelů)
- 12 funkčních tlačítek a trimer pro ladění jasu displeje
- Komunikace přes Bluetooth, dosah maximálně 100m
- Rychlost přenosu dat 57,6 kb/s
- Zobrazení aktuálního stavu připojení se serverem na základní obrazovce
- Posílání a příjem zprávy s maximální délkou 160 znaků
- Zobrazení na displeji, kolik bylo přijato nových textových zpráv
- Paměť pro maximálně 20 přijatých zpráv, která se po jejich přečtení uživatelem opět uvolní
- DIP přepínač pro základní nastavení Bluetooth transceiveru
- Automatické připojení k serveru po ztrátě signálu, případně vypnutí a zapnutí modulu.

4.3.1 Popis ovládání

Po spuštění informačního modulu (připojení k napájecímu napětí 3,0V-3,3V), naběhne na displeji základní obrazovka, jejíž vzhled můžete vidět na obrázcích 4-4 a 4-5. Je na ní zobrazen název daného modulu (BTM2 – Bluetooth Modul 2) a aktuální stav.

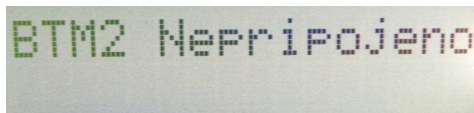
V případě, že je zařízení propojeno se serverem, obrázek 4-4, je na displeji zobrazen stav „Připojeno“. V tomto stavu může informační modul přijímat, číst a posílat zprávy. Pokud chcete psát novou zprávu, stisknete první tlačítko na klávesnici. Ostatní klávesy nemají v tomto stavu žádnou funkci.



Obrázek 4-4: Základní obrazovka, připojený stav

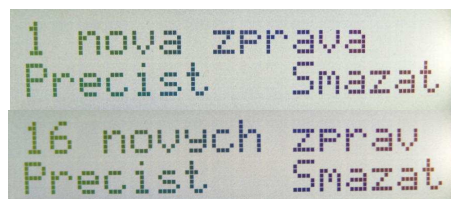
Pokud informační modul nemá spojení se serverem, obrázek 4-5, je na displeji zobrazen stav Nepřipojeno. V tomto stavu nelze s informačním modulem nic dělat. Je třeba počkat, až s ním server automaticky naváže spojení, to poznáte přepsáním základní obrazovky do stavu Připojeno (obrázek 4-4).

Pokud je od serveru přijata nová zpráva, zobrazí se na displeji informace o přijaté zprávě, viz obrázek 4-6. V případě, že budou přicházet další zprávy dříve, než si přijatou zprávu přečtete, budete



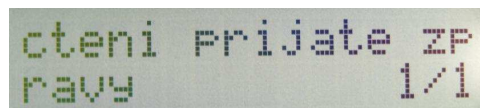
Obrázek 4-5: Základní obrazovka, nepřipojený stav

o tom informování číslem, udávajícím kolik nových zpráv bylo přijato. Informační modul ovšem umožňuje přijmout najednou maximálně 20 nových zpráv. Od doby, kdy je na displeji zobrazen stav „20 nových zpráv“, nebudou další zprávy poslané serverem přijaty. Přijaté zprávy můžete přečíst stisknutím první klávesy, případně je smazat stisknutím třetí klávesy. Další klávesy zde nemají žádnou funkci. Pokud je počet přijatých zpráv větší než 1, budou při stisknutí klávesy Smazat nenávratně odstraněny všechny nově přijaté zprávy.



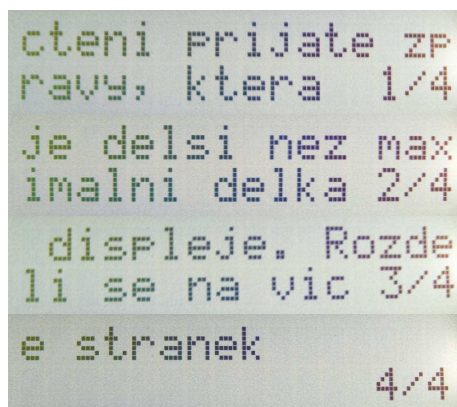
Obrázek 4-6: Nové přijaté zprávy

Po stisknutí klávesy Přečíst se dostanete režimu čtení zprávy. Jak vypadá zobrazení přijaté zprávy na displeji, můžete vidět na obrázku 4-7. Pro zobrazení obsahu zprávy je použito 28 znaků displeje z 32 dostupných. Poslední 4 znaky jsou použity pro zobrazení počtu stránek čtené zprávy a index aktuální stránky, která je právě zobrazena (přijatá zpráva na obrázku 4-7 je kratší než 28 znaků, proto je zobrazen stav 1/1).



Obrázek 4-7: čtení zprávy

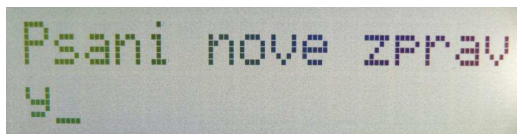
Jak vypadá čtení zprávy, která je rozdělena na více stránek (obrazovek), je vidět na obrázku 4-8. Taková zpráva je rozdělena na části po 28 znacích. Mezi těmito stránkami přecházíte stisknutím páté nebo jedenácté klávesy, kde jedenáctou klávesou se posouváte ve zprávě směrem dolů a pátou klávesou směrem nahoru. V případě, že v době čtení zprávy je od serveru přijata další, zařadí se na konec fronty přijatých zpráv. Stisknutím první klávesy přejdete na čtení další přijaté zprávy v pořadí, případně na základní obrazovku, pokud žádná další zpráva nenásleduje.



Obrázek 4-8: čtení zprávy, která je delší než 32 znaků

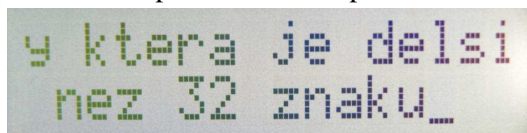
4.3.2 Psaní zprávy:

Jak již bylo zmíněno výše, do režimu psaní nové zprávy se dostanete stisknutím klávesy Zpráva na hlavní obrazovce. Na displeji se objeví blikající kurzor a využitím klávesnice můžete psát zprávu. Všechny znaky se píšou pomocí devíti kláves. Přiřazení znaků k jednotlivým klávesám je znázorněno níže. Mezi jednotlivými znaky, nacházejícími se na jedné klávese přecházíte jejím vícenásobným stisknutím. Aktuální znak je ihned zobrazován na pozici kurzoru, kdy při následném stisknutí stejné klávesy je přepsán následujícím znakem. V momentě, kdy je stisknuta jiná klávesa nebo vyprší časový limit pro následný stisk (cca. 500ms), přejde kurzor na další pozici a předchozí znak je zapsán napevno.



Obrázek 4-9: psaní zprávy

Pokud je psaná zpráva delší než 32 znaků, přejde kurzor na další prázdný řádek a původně zobrazovaný horní řádek nebude viditelný na displeji. Tento průběh je znázorněn na obrázcích 4-9 a 4-10. Pokud znaky mažete, probíhá tento posuv řádků naopak.



Obrázek 4-10: psaní zprávy delší než 32 znaků

Po odeslání zprávy se dostanete na hlavní obrazovku. V případě, že vám bude ze serveru zaslána zpráva v době, kdy píšete zprávu, zobrazí se na displeji obrazovka informující o nově přijaté zprávě (obrázek 4-6) a poté co si zprávu/zprávy přečtete nebo smažete, budete automaticky pokračovat v psaní zprávy přesně v místě, kde jste skončili, před přijetím zprávy.

4.3.3 Využití kláves při psaní zprávy:

1. **klávesa** – Odeslání zprávy na server (nelze odeslat prázdnou zprávu).
2. **klávesa** – Přepínání mez psaním malých/velkých písmen.
3. **klávesa** – Smazání posledního znaku zprávy.
4. **klávesa** – Psaní znaků: „ “, „. “, ?, !, 1, 0
5. **klávesa** – Psaní znaků: a(A), b(B), c(C), 2
6. **klávesa** – Psaní znaků: d(D), e(E), f(F), 3
7. **klávesa** – Psaní znaků: g(G), h(H), i(I), 4
8. **klávesa** – Psaní znaků: j(J), k(K), l(L), 5
9. **klávesa** – Psaní znaků: m(M), n(N), o(O), 6
10. **klávesa** – Psaní znaků: p(P), q(Q), r(R), s(S), 7
11. **klávesa** – Psaní znaků: t(T), u(U), v(V), 8
12. **klávesa** – Psaní znaků: w(W), x(X), y(Y), z(Z), 9

4.3.4 DIP Switch

Na desce plošného spoje vedle klávesnice je umístěn DIP přepínač, který slouží pro nastavení Bluetooth modulu [11]. Ten je zde umístěn spíše jako vývojový nástroj a pro běžné užití modulu jej nechte ve stavu 4*OFF. Použití jeho jednotlivých přepínačů je uvedeno v tabulce 4-1.

Funkce	DIP Switch	Nastavení (OFF = GND / ON = 3,3V)
Factory Reset	1	OFF = vypnuto, ON = aktivní
Auto Discovery / Pairing	2	OFF = vypnuto, ON = zapnuto
Auto-Connect	3	OFF = vypnuto, ON = zapnuto
Baudrate	4	OFF = Nastavené (57,6K) ON=9600

Tabulka 4-1: Použití DIP přepínače

Factory Reset

Přepnutím do aktivního stavu během zapínání modulu dojde ke spuštění resetovacího stavu. Poté přepnutím ON/OFF třikrát za sebou způsobíte reset všech nastavení na factory defaults.

Auto Discovery/Pairing Mode

používá se společně s přepínačem číslo 3 (Auto-Connect). Pokud je přepínač č. 3 také nastaven, provede Inquiry Scan, kde hledá zařízení ve speciální shodující se třídě (0x55AA) a pokud ho najde, uloží si jeho adresu a poté se k němu automaticky připojí.

Pokud přepínač č. 3 není nastaven, přepne se do Slave módu a bude čekat, až bude nalezeno Masterem ze speciální skupiny.

Auto Connect mode

Pokud je nastaveno, automaticky se připojí na uloženou adresu.

Baud Rate select

Používá se pro volbu baudrate mezi 9600 a hodnotou nastavenou uživatelem (factory defaults 115200, v této práci je použita hodnota 57600). Postup pro změnu hodnoty baudrate na jinou je popsán v manuálu k Bluetooth modulu[.]

Poznámka: Stav přepínačů je snímán pouze během zapínání modulu, během prvních 500ms.

5 Implementace aplikace pro server

Serverová aplikace je psána v jazyce C++. Vyvíjena byla v programu Qt Creator 1.3.1, založeném na Qt 4.6.2 [14]. Pro Bluetooth komunikaci přes sériový port, byla navíc použita nadstandardní knihovna `qextserialport.h`.

5.1 QExtSerialPort

Knihovna `qextserialport` [15] je rozšiřující knihovna do Qt, která umožní práci se sériovými porty COM jak v OS Windows, tak na UNIXových systémech.

5.1.1 Použité metody z této knihovny

```
QextSerialPort(const QString & name, QueryMode mode =  
QextSerialPort::Polling)
```

Vytvoří nový sériový port se jménem `name` a řízený podle hodnoty parametru `Polling`.

```
virtual void setBaudRate(BaudRateType)
```

Nastaví rychlost přenosu pro daný sériový port, podle hodnoty předané parametrem `BaudRateType`.

```
virtual void setFlowControl(FlowType)
```

Zapne nebo vypne řízení toku (handshake), pro daný sériový port, podle hodnoty předané parametrem `FlowType`.

```
virtual setParity(ParityType)
```

Nastaví paritu pro daný sériový port, podle hodnoty předané parametrem `ParityType`.

```
virtual setDataBits(DataBitsType)
```

Nastaví počet datových bitů pro přenos dat u daného sériového portu, podle hodnoty předané parametrem `DataBitsType`.

```
virtual setStopBits(StopBitsType)
```

Nastaví počet stop bitů pro přenos dat u daného sériového portu, podle hodnoty předané parametrem `StopBitsType`.

```
virtual bool open(OpenMode mode)
```

Otevře daný sériový port v módu, předaný parametrem `mode`.

```
virtual void close()
```

Zavře daný sériový port.

```
virtual qint64 readData(char * data, qint64 maxSize)
```

Přečte data z bufferu pro daný sériový port a uloží je do proměnné `data`. Přečte maximálně `maxSize` bytů.

```
virtual qint64 writeData(const char * data, qint64 maxSize)
```

Zapíše obsah proměnné `data` do bufferu pro daný sériový port, kde se poté automaticky odešlou. Zapíše maximálně `maxSize` bytů.

5.2 Implementované třídy

V rámci aplikace pro server byly implementovány třídy:

MainWindow – třída, která zajišťuje obsluhu hlavního okna aplikace

Dialog – obsluha sekundárního okna s nastavením

Thread – automatické připojování informačních modulů, které jsou nastaveny a jsou v dosahu

5.2.1 Některé metody těchto tříd

```
void MainWindow::readData(int device)
```

Metoda čte data z bufferu určitého COM portu (Informačního modulu), který je upřesněn parametrem `device` a ty zapíše do textového boxu v hlavním okně aplikace spolu s aktuálním časem a příjemcem, kterému je určena. Tato metoda je volaná pouze jako reakce na událost `readyRead()`, která, která je spouštěna v případě, že na buffer sériového portu přišla data.

```
void MainWindow::sendData()
```

Metoda pošle zprávu, kterou uživatel napsal do řádky „zpráva k odeslání“, na zařízení zvolené v rolovací nabídce „Komu:“. Poté smaže z řádky odeslanou zprávu a přepíše jí do textového boxu spolu s aktuálním časem a příjemcem, kterému byla určena.

```
void MainWindow::LoadSettings()
```

Načte nastavení ze souboru `settings.xml`, kde je uloženo s kolika zařízeními má server pracovat a čísla portů, na kterých se nachází (na které byla spárována).

```
void MainWindow::timerConnect()
```

Metoda periodicky volaná od časovače. Nastavený čas závisí na počtu aktivních zařízení v nastavení aplikace.

```
void Dialog::SaveSettings()
```

Uloží nastavení, které uživatel provedl v okně dialog, do souboru `settings.xml`.

```
bool Thread::testCommunication(int device)
```

Metoda, která se spouští jako další thread, testuje všechny informační moduly, které jsou nastaveny v nastavení a provádí test komunikace. Odešle 1 byte na modul a poté se ho pokusí přijmout zpět. Pokud se to nezdaří, snaží se k němu znovu připojit.

5.3 Popis aplikace

Aplikace Bluetooth terminal je určena pro serverovou část (PC). Slouží pro příjem a zasílání zpráv na vytvořené informační moduly. Je možné jí použít až pro 8 informačních modulů, které byly předem spárovány s počítačem, na kterém se bude Bluetooth terminal spouštět. Moduly, které byly zvoleny v nastavení, připojí k serveru, pokud jsou zapnuty a v dosahu. Pokud některý informační modul nebyl po spuštění připojen, pokouší se s ním v pravidelných intervalech znovu navázat spojení. Pravidelně také testuje funkčnost komunikace mezi všemi připojenými moduly. Nabídka s výběrem adresáta zprávy je aktualizována a nachází se v ní pouze ty moduly, které jsou připojeny, a test komunikace u nich proběhl správně.

5.3.1 Ovládání

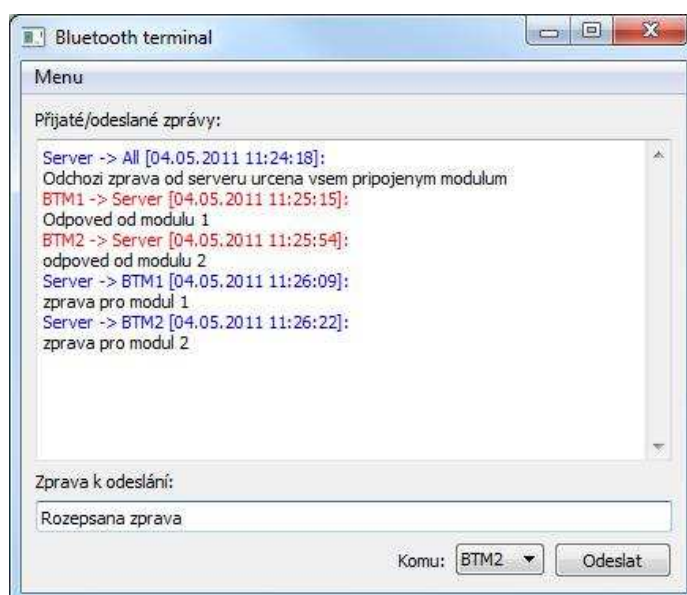
Po spuštění se program připojí k dostupným informačním modulům. Ty se poté objeví ve spodní části okna, v rolovací nabídce Komu : . Jestliže se nezdaří připojení ani k jednomu modulu, bude tam pouze nápis „Žádný“. V případě, že se připojí více než jedno zařízení, objeví se ve výběru možnost „všem“. Rolovací nabídka po připojení dvou modulů je zobrazena na obrázku 5-1.



Obrázek 5-1: Výběr odesílatele

Odesílání zprávy:

Vaši zprávu, kterou budete chtít odeslat, napíšete do řádku "Zpráva k odeslání", viz obrázek 5-2. Poté vyberte, kterému modulu chcete zprávu doručit, případně zvolte Všem, čímž zprávu odešlete všem připojeným modulům. Nakonec stisknete tlačítko odeslat nebo stisknete klávesu enter.



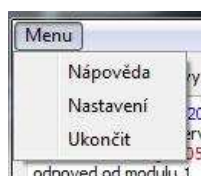
Obrázek 5-2: Hlavní okno aplikace

Přijaté zprávy:

Zprávy, které obdržíte od informačního modulu, se zobrazí ve velkém textovém boxu "přijaté/odeslané zprávy". Zobrazuje se zde také, od kterého modulu byla zpráva přijata a přesný čas doručení. Hlavička přijaté zprávy je vtištěna červeně a hlavička zprávy odeslané z této aplikace (ze serveru) je vtištěna modře, viz obrázek 5-2.

Menu:

V horní části okna se nachází menu, přes které můžete spustit nápovědu, nastavení, případně ukončit program, viz obrázek 5-3.

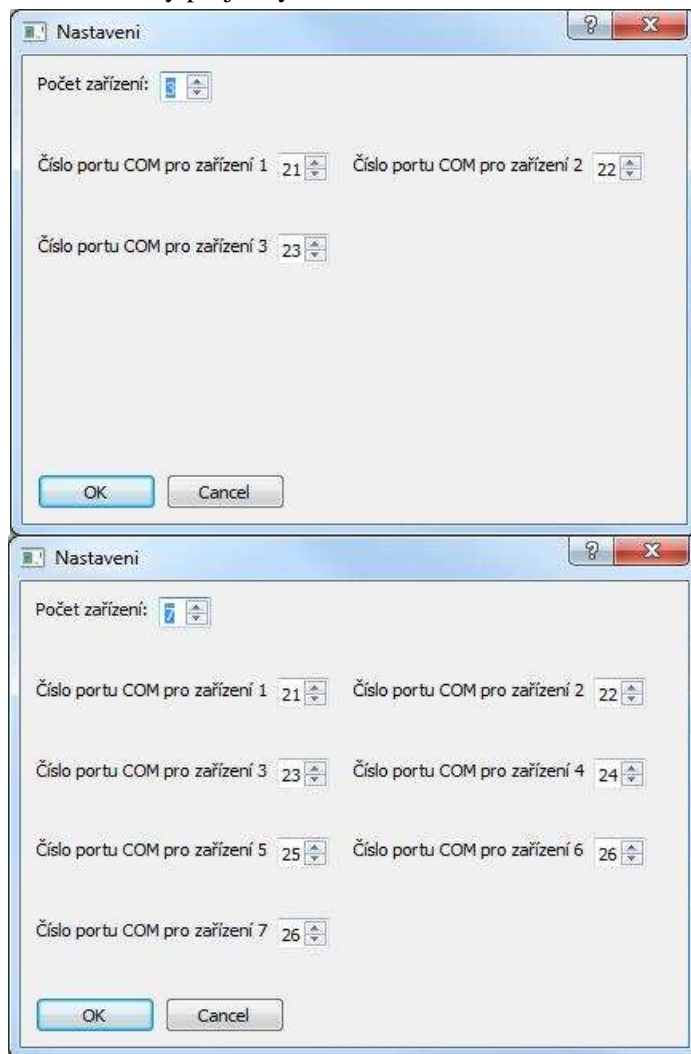


Obrázek 5-3: Menu

Nastavení:

V nastavení pod položkou „Počet zařízení:“, zvolíte počet informačních modulů, se kterými chcete komunikovat. Zadává se číslo v rozsahu od 1 do 8. Podle zadaného čísla se zobrazí stejný počet polí pro přiřazení čísla COM portu k jednotlivým zařízením, to můžete vidět na obrázku 5-4.

Po provedení požadovaných změn je uložíte stisknutím tlačítka OK, případně ponecháte původní nastavení stisknutím tlačítka Cancel. Po úpravách v nastavení je třeba celou aplikaci restartovat, aby se provedené změny projevíly.



Obrázek 5-4: Nastavení - více zařízení

6 Závěr

V této práci byl čtenář v první kapitole seznámen s použitou technologií Bluetooth. V následujících dvou kapitolách bylo popsáno vyvíjení Informačních modulů od návrhu, přes jejich výrobu, až po implementaci firmware. V 5. Kapitole byla popsána aplikace, která slouží pro serverovou část, včetně použitých knihoven.

Výsledné Informační moduly byly vytvořeny s důrazem na mobilitu s možností napájení z baterií. Úspěšně otestován byl provoz na 2 AA baterie v sériovém zapojení. Ovšem v budoucnu by měly být provozovány s nabíjecím akumulátorem a nejspíše bude potřeba doplnit i stabilizátor napětí. Při návrhu firmware pro řídicí mikrokontrolér byla vyvinuta snaha o co možná nejvíce intuitivní ovládání i přes omezující faktory, kterými byly malý displej a ovládání pomocí klávesnice s dvanácti tlačítky. I přes handicap displeje s 32 znaky, jsou podporovány zprávy do délky 160 znaků, kdy při čtení došlých zpráv může uživatel libovolně přecházet mezi jednotlivými stránkami zprávy a při psaní dochází k automatickému posuvu o 1 řádek směrem dolů nebo nahoru, podle toho, zda jsou znaky psány nebo mazány. Byla také vytvořena podpora pro více přijatých zpráv, kde je jejich počet zobrazen na displej, a při čtení jsou řazeny za sebe.

Serverová aplikace byla vyvíjena s důrazem na přehlednost a jednoduchost ovládání. Podporuje komunikaci až s 8 informačními moduly současně, otestován byl však pouze provoz se dvěma. Aplikace provádí připojování a odpojování nastavených zařízení automaticky a ty komunikující jsou zobrazeny v nabídce odesílatelů. Program má také možnost odeslat zprávu všem aktivním zařízením. Všechny přijaté a odeslané zprávy jsou ukládány do okna, takže uživatel může vidět veškerou komunikaci od zapnutí aplikace. Přijaté a odeslané zprávy jsou pro přehlednost barevně odlišeny. Uživatelské nastavení je ukládáno do XML souboru.

Mezi další rozšíření do budoucna by patřila umístění signalizačního prvku na informační moduly, který by oznamoval příjem nové zprávy (např. reproduktor) a zprovoznění vypínání displeje po určité době nečinnosti. To nyní nefunguje z důvodu, že při původním návrhu s ním nebylo počítáno a napájení displeje je na DPS přivedeno přímo ze zdroje napájení.

Celý systém byl otestován na OS Windows 7 32bit a Windows XP. Důvodem pro jeho vznik bylo vyzkoušet si práci s Bluetooth a programování mikrokontrolérů řady AVR v praxi. Celý systém není příliš perspektivní, jelikož v dnešní době již existuje plno podobných vyspělejších řešení s vyšším dosahem, jejichž ceně se nedá při kusové výrobě konkurovat.

7 Literatura

- [1] Bluetooth. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2010-12-31]. Dostupné z WWW: <http://en.wikipedia.org/wiki/Bluetooth>
- [2] Jiří Svoboda: *Principy a perspektivy technologie Bluetooth*, Sdělovací technika 08/2004.
- [3] Jiří Čánský: *Mobilní komunikace Bluetooth*. Prosinec 2010. Dostupné na URL: http://radio.feld.cvut.cz/personal/mikulak/MK/MK06_semestralky/Bluetooth_CanskyJ.pdf
- [4] Bluetooth. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2010-12-31]. Dostupné z WWW: http://en.wikipedia.org/wiki/Bluetooth_protocols
- [5] Bray, J., Thurman, Ch.: *Bluetooth: Connect Without Cables*. Prentice-Hall, Inc. Upper Saddle River, 2001.
- [6] *How stuff works* [online]. 2005 [cit. 2011-05-07]. How Bluetooth Works. Dostupné z WWW: <http://electronics.howstuffworks.com/bluetooth.htm>.
- [7] *Palo wireless* [online]. 2006 [cit. 2011-05-07]. RFCOMM Protocol. Dostupné z WWW: <http://www.palowireless.com/infotooth/tutorial/rfcomm.asp>.
- [8] Atmel Corporation: *ATmega164A/164PA/324A/324PA/644A/644PA/1284/1284P Datasheet*. Únor 2011. Dostupné na URL: http://atmel.com/dyn/resources/prod_documents/doc8272.pdf
- [9] Newhaven: *NHD-0216HZ-FSW-FBW-3V3C Datasheet*. Únor 2011. Dostupné na URL: <http://www.newhavendisplay.com/specs/NHD-0216HZ-FSW-FBW-3V3C.PDF>
- [10] Roving networks: *RN-41 Datasheet*. Únor 2011. Dostupné na URL: <http://www.rovingnetworks.com/Docs/Bluetooth-RN-41-DS.pdf>
- [11] Roving networks: *RN-41 User manual*. Únor 2011. Dostupné na URL: <http://www.rovingnetworks.com/Docs/Bluetooth-RN-UM.pdf>
- [12] Atmel Corporation: *AVR Studio 4*. Leden 2011. Dostupné na URL: http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725
- [13] ASIX: *UŽIVATELSKÝ MANUÁL pro PRESTO*. Duben 2011. Dostupné na URL: http://www.asix.cz/download/programmers/programmers_cz.pdf
- [14] Trolltech: *Qt Reference Documentation*. Duben 2011. Dostupné na URL: <http://doc.trolltech.com/4.6/index.html>

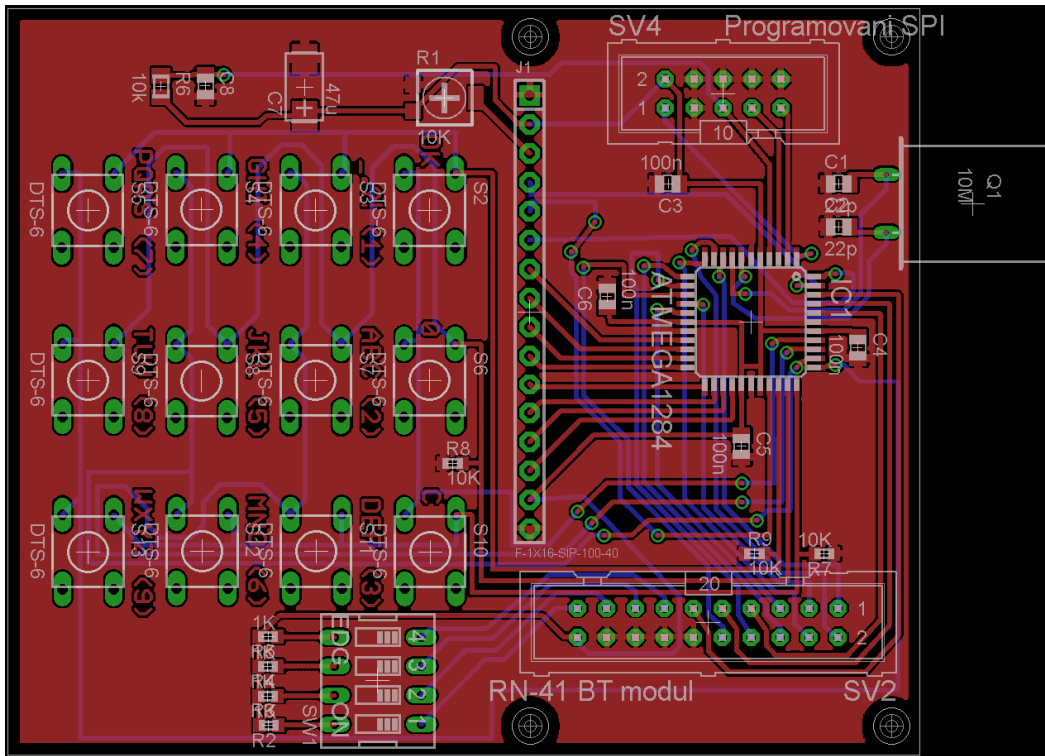
- [15] Source forge: *QextSerialPort*.
Květen 2011. Dostupné na URL: <<http://qextserialport.sourceforge.net>>
- [16] *Květákov* [online]. 2006 [cit. 2011-05-08]. Články – AVR.
Dostupné na URL: <<http://www.kvetakov.net/clanky/avr/>>
- [17] *ROOT* [online]. 2006 [cit. 2011-05-08]. Externí sériové sběrnice SPI a I²C.
Dostupné na URL: <<http://www.root.cz/clanky/externi-seriove-sbornice-spi-a-i2c/>>

Seznam příloh

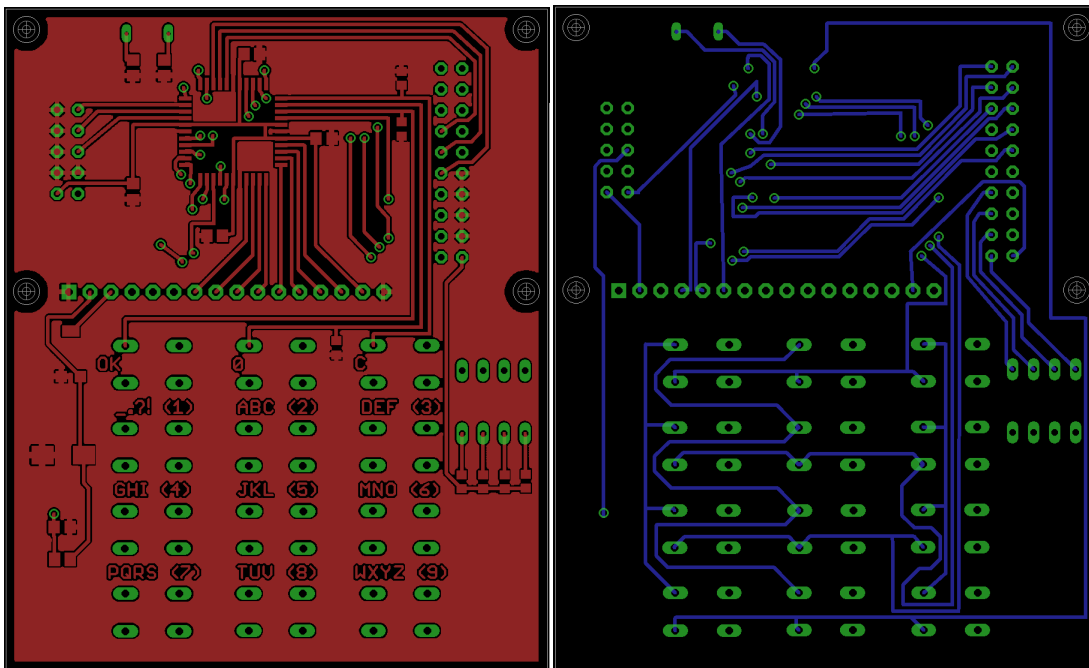
- Příloha 1 - Deska plošného spoje, seznam osazených součástek a zapojení konektorů.
- Příloha 2 - Fotografie vyrobených Informačních modulů
- Příloha 3 - Obsah CD

Příloha 1

Deska plošného spoje Informačního modulu:



DPS zobrazení i se součástkami a popisky



Horní (vlevo) a spodní strana (vpravo) DPS

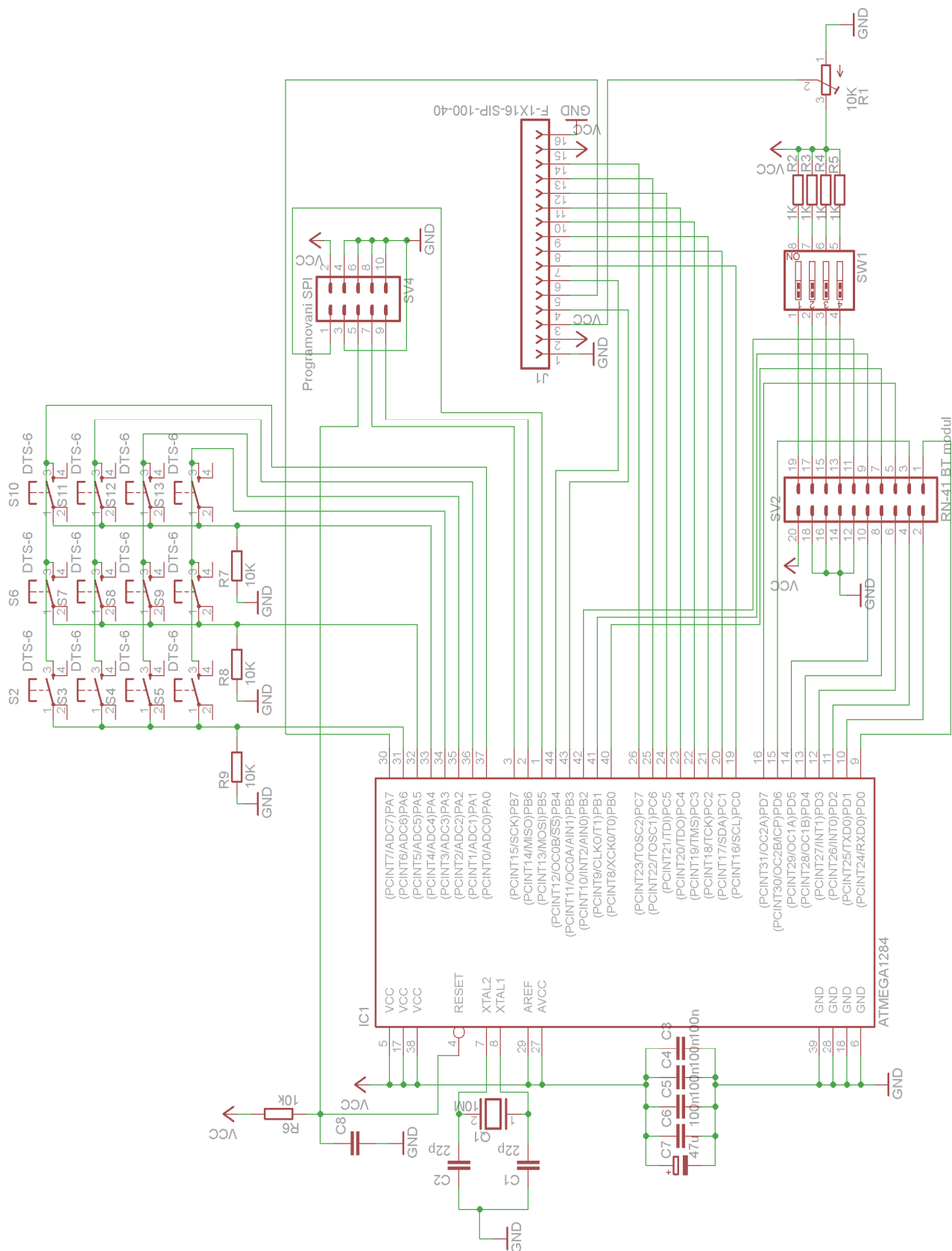


Schéma informačného modulu

Výpis použitých součástek na osazení DPS:

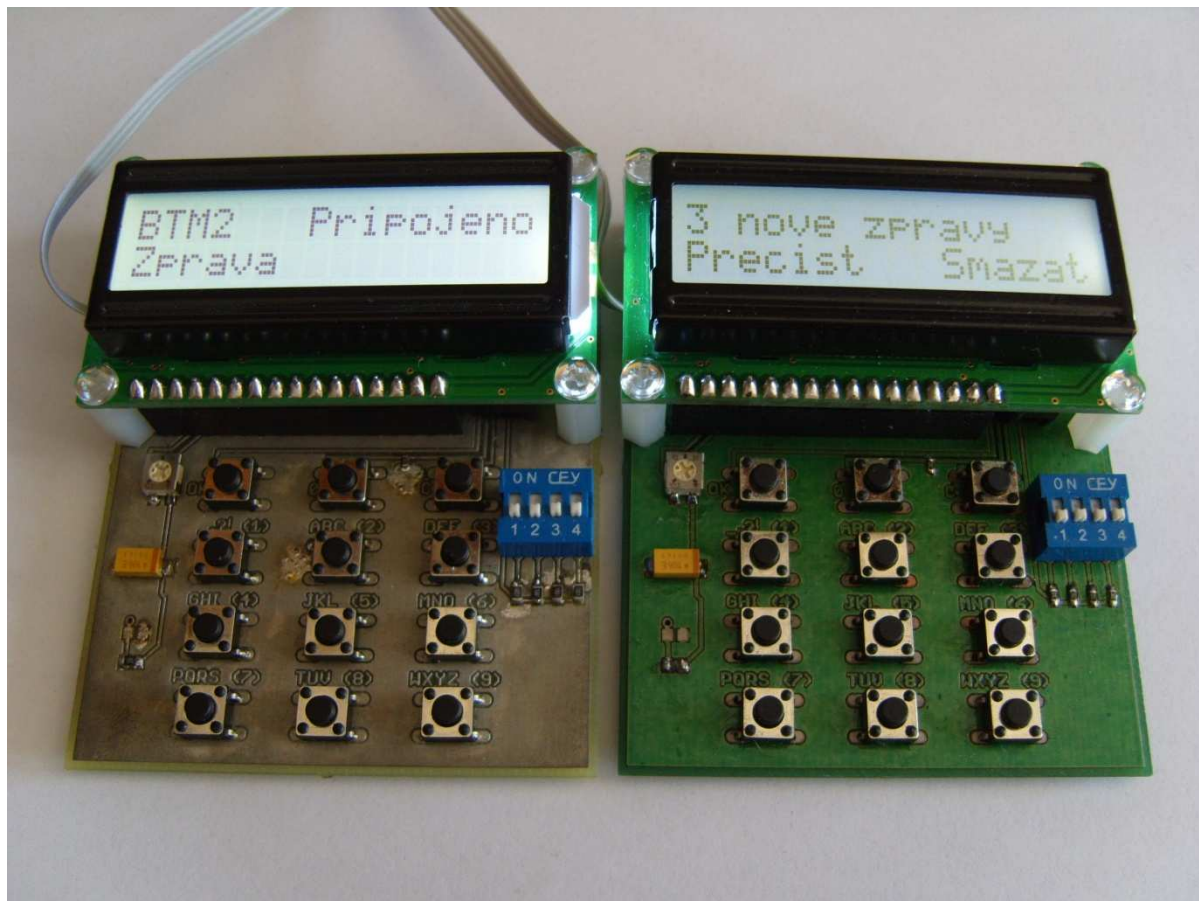
Part	Value	Device	Package	Library
C1	22p	C-EUC0805	C0805	rcl
C2	22p	C-EUC0805	C0805	rcl
C3	100n	C-EUC0805	C0805	rcl
C4	100n	C-EUC0805	C0805	rcl
C5	100n	C-EUC0805	C0805	rcl
C6	100n	C-EUC0805	C0805	rcl
C7	47u	CPOL-EUCT6032	CT6032	rcl
C8		C-EUC0805	C0805	rcl
IC1	ATMEGA1284	ATMEGA644A	TQFP44	atmel
J1	F-1X16-SIP-100-40	F-1X16-SIP-100-40	SIP-100-16-40	con-headers-jp
Q1	10M	XTAL	Q	special
R1	10K	R-TRIMM4G/J	RTRIM4G/J	rcl
R2	1K	R-EU_R0603	R0603	rcl
R3	1K	R-EU_R0603	R0603	rcl
R4	1K	R-EU_R0603	R0603	rcl
R5	1K	R-EU_R0603	R0603	rcl
R6	10k	R-EU_M0805	M0805	rcl
R7	10K	R-EU_R0603	R0603	rcl
R8	10K	R-EU_R0603	R0603	rcl
R9	10K	R-EU_R0603	R0603	rcl
S2	DTS-6	DTS-6	DTS-6	switch-tact
S3	DTS-6	DTS-6	DTS-6	switch-tact
S4	DTS-6	DTS-6	DTS-6	switch-tact
S5	DTS-6	DTS-6	DTS-6	switch-tact
S6	DTS-6	DTS-6	DTS-6	switch-tact
S7	DTS-6	DTS-6	DTS-6	switch-tact
S8	DTS-6	DTS-6	DTS-6	switch-tact
S9	DTS-6	DTS-6	DTS-6	switch-tact
S10	DTS-6	DTS-6	DTS-6	switch-tact
S11	DTS-6	DTS-6	DTS-6	switch-tact
S12	DTS-6	DTS-6	DTS-6	switch-tact
S13	DTS-6	DTS-6	DTS-6	switch-tact
SV2	RN-41	ML20	ML20	con-ml
SV4	Programovani SPI	ML10	ML10	con-ml
SW1	SW_DIP-4	EDG-04	-	special

Propojení IO Atmaga644pa:

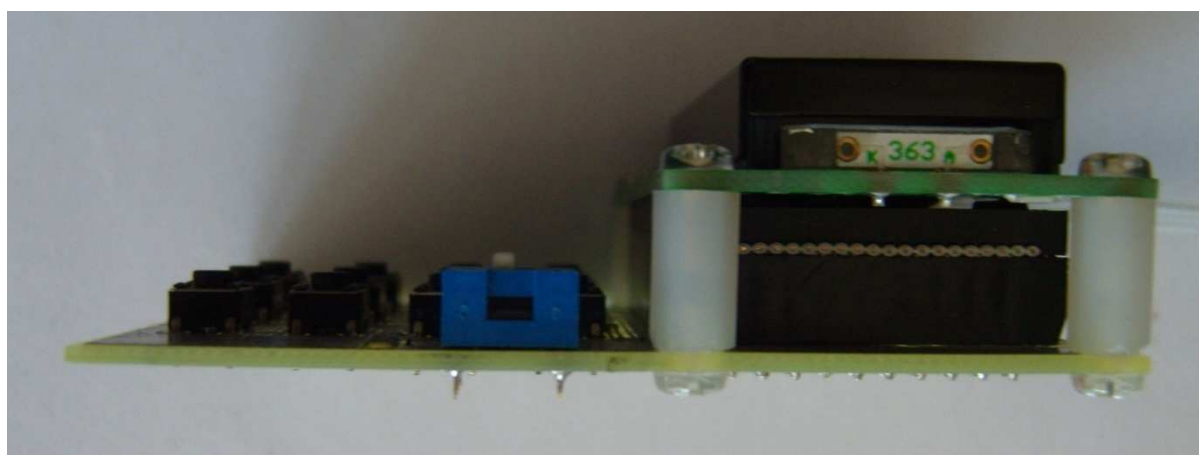
PIO	pin - konektor	pin - zařízení	popis
PA0	-	COL2	Klávesnice - sloupec 2
PA1	-	ROW1	Klávesnice - radek 1
PA2	-	COL1	Klávesnice - sloupec 1
PA3	-	ROW4	Klávesnice - radek 4
PA4	-	COL3	Klávesnice - sloupec 3
PA5	-	ROW3	Klávesnice - radek 3
PA6	-	ROW2	Klávesnice - radek 2
PA7	J1-5	RS	Display - register select
PB0	SV2-7		Rezerva
PB1	SV2-9		Rezerva
PB2	SV2-11		Rezerva
PB3	J1-4	R/W	Display - Read/Write signal
PB4	J1-6	E	Display - operation enable signal
PB5	SV4-1	MOSI	Programování MCU
PB6	SV4-9	MISO	Programování MCU
PB7	SV4-7	SCK	Programování MCU
PC0	J1-7	DB0	Display - data
PC1	J1-8	DB1	Display - data
PC2	J1-9	DB2	Display - data
PC3	J1-10	DB3	Display - data
PC4	J1-11	DB4	Display - data
PC5	J1-12	DB5	Display - data
PC6	J1-13	DB6	Display - data
PC7	J1-14	DB7	Display - data
PD0	SV2-1	UART_TX	RN-41 - sériová komunikace
PD1	SV2-2	UART_RX	RN-41 - sériová komunikace
PD2	SV2-4	PIO2	RN-41 - status
PD3	SV2-6	PIO5	RN-41 - status
PD4	SV2-8	PIO8	RN-41 -status
PD5	SV2-10	PIO9	RN-41 - IO
PD6	SV2-3	PIO10	RN-41 - IO
PD7	SV2-5	PIO11	RN-41 - IO

Příloha 2

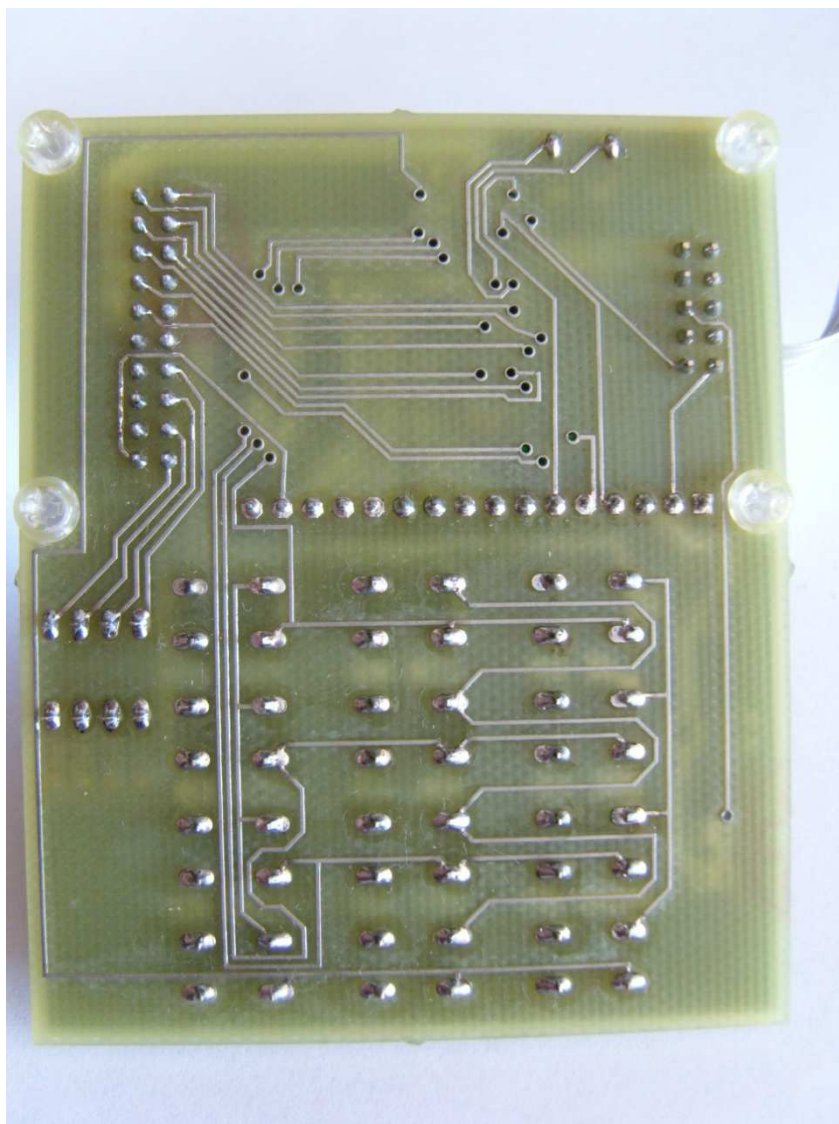
Fotky Informačních modulů



Informační moduly - pohled shora



Informační modul - boční pohled



Informační modul - spodní pohled

Příloha 3

Obsah CD

- Adresář Dokumentace – obsahuje bakalářskou práci ve formátu PDF a DOCX.
- Adresář Atmega – obsahuje zdrojový soubor programu mikrokontroléru v jazyce C.
- Adresář BTterminal – obsahuje zdrojové soubory v jazyce C++ pro Qt. V podřazeném adresáři BIN se nachází spustitelný soubor přeložený pro windows společně s knihovnami potřebnými pro jeho běh (otestováno na win7 32 bit a win XP sp3).
- Adresář DPS – obsahuje schéma a návrh DPS informačního modulu pro Eagle 5.11.0.
- Adresář Fotky – obsahuje fotografie vytvořených informačních modulů.